SIEMENS

# 9750 Data Display Terminal

# TRANSDATA

## User's Guide

# 9750 Data Display Terminal TRANSDATA®

# User's Guide

March 1983

Contents

# 1 General information on the 9750 Data Display Terminal and its integration into teleprocessing systems

## 1.1 9750 Data Display Terminal

The 9750 Data Display Terminal is a component of the TRANSDATA 810 Terminal System and can be used in dialog mode or, for data entry applications, in dialog and field mode.

The 9750 can be operated as a

- standalone terminal or, with the appropriate facilities, as part of a
- clustered configuration.

A standalone 9750 can connect to a host computer either directly or via data communication equipment such as modems, concentrators and interface expanders.

Clustered 9750's connect to 8170 or 8171 Cluster Controllers or to 966x Terminal Computers via the BAM interface controller.

The 9750 comprises a display unit (CRT plus single integrated controller) and a keyboard. Up to 1920 characters of user data can be displayed - in a 24 lines by 80 characters format - in green against a dark background. Line 25 is reserved for statuses and system messages.

### 1.1.1 Compatibility and system integration

The message formats and function control capability of the 9750 Data Display Terminal, which can be used in conjunction with the Siemens operating systems PDN, BS1000 and BS2000, gear the device to TRANSDATA 960, System 7.500 and System 7.700 environments. The 9750 is compatible with the 8160 Data Display Terminal in all hardware and software configurations.

Field mode can be implemented only in a TRANSDATA 960 environment in conjunction with the software components TRINIDAD and BERMUDA.

The keyboard assignment corresponds to that on the 8160 Data Display Terminal.

### 1.1.2 Operating modes

Dialog mode

In dialog mode, data entry is performed via the keyboard; transmission to the host computer is initiated by activating transmit functions.

Data transmission may be formatted or non-formatted. In formatted mode, the system outputs format commands to the 9750, specifying field attributes (protected, numeric etc.) and display attributes (reduced intensity, flashing etc). In non-formatted mode, neither the system nor the terminal user is tied to specified text formatting or presentation arrangements. (NB: If field separators are output, the 9750 switches over automatically to formatted mode.)

Field mode

This mode is used for mask-supported data entry and permits the following:

- data entry and
- keyboard crosschecking of entered data.

In both instances, the 9750 must be connected to a terminal computer. Data entry is supported by the software services BERMUDA and TRINIDAD via the PDN operating system executing in the terminal computer.

Operator prompting by the host is thorough: it releases only one field at a time for manipulation, and the cursor is positioned at the start of the field. When the field is full, it is sent to the host, where it undergoes validity-checking. If the result proves positive, the next field is released. If the result proves negative, the entire screen starts flashing.

## 1.2 Further components of the TRANSDATA 810 Terminal System

### 1.2.1 Cluster controllers

Cluster controllers permit a number of locally concentrated terminals to be connected cost-effectively to a computer system. Either the 8170 Cluster Controller (local) or the 8171 Cluster Controller (remote) is used, depending on the distance involved.

Four terminals can connect to a cluster controller in its basic configuration; further expansion options permit a total of 32 terminals to be attached. The following terminals can be attached:

9750 Data Display Terminals;
8160, 8161 and 8162 Data Display Terminals;
8112 Printer Terminals;
90037-xxx Printers.

A data display terminal and a printer may intercommunicate without placing a burden on either the transmision path to the host or on the host itself.

Information on the maximum line length between a terminal and a cluster controller can be found in the relevant technical documentation available from Siemens AG.

#### 1.2.1.1 8170 Cluster Controller (local)

The 8170 Cluster Controller (local) is used to group the data traffic for a number of local terminals onto the byte multiplex channel of a Siemens System 7.500, 7.700 or 4004 host computer. It may be configured up to 30 m away from the host. For more detailed information, please refer to the relevant technical documentation available from Siemens AG.

#### 1.2.1.2 8171 Cluster Controller (remote)

The 8171 Cluster Controller (remote) is used to group the data traffic for a number of terminals on either one or two remote lines to the host computer. A freestanding modem or an integrated baseband modem may be used as data transmission facilities. The maximum transmission rate for transmission to host is 9600 bps, and the communication protocol used is MSV1.

The lines can be operated as a line group, thus enhancing line utilization. This option is especially useful in order to keep wait times to a minimum when data traffic is high. For more detailed information, please refer to the relevant technical documentation available from Siemens AG.

### 1.2.2 8112 Printer Terminal

The 8112 Printer Terminal comprises a printer terminal controller and either one or two printers. The

The 8112 can be configured, in conjunction with a cluster controller, as

- a shared printing system for all data display terminals connected to a cluster controller,
- shared printing system for specific data display terminals connected to a cluster controller, or as the
- system printer for the entire DP system.

It is still possible to operate the printer terminal as a standalone device, too.

### 1.2.3 97507 Integrated Manual Badge Reader

As an optional feature, a manual badge reader capable of reading SIPASS and ABA-format badges may be built into the 9750 Data Display Terminal's keyboard.

## 1.3 Attachable printers

The following printers can connect to the 9750 Data Display Terminal:

| Printer | 8121 *) | 8122 | 8124 | 9002 | 9003 |
|---|---|---|---|---|---|
| Technology | Ink jet | Needle or ink jet | Needle | Ink jet | Ink jet |
| Max. print speed | 180 cps | 90 cps | 90 cps | 270 cps | 250 cps |
| Chars. per line | 132 | 80 | 80 | 80 | 136 |
| Paper | Margin-punched | Margin-punched | Margin-punched | Margin-punched | Margin-punched |
| Paper guide | 1 or 2 cont.-form guides, or single-sheet guide, or single-sheet guide and 1 cont.-form guide | 1 cont. form guide | Cont.-form and single-sheet guide | 1 cont.-form or single-sheet guide | 1 or 2 cont.-form guides, single-sheet guide |
| Fonts | Standard<br>Wide | Standard<br>Standard, italics<br>Large caps<br>Large caps, italics | Standard<br>Standard, italics<br>Condensed<br>Condensed, italics<br>Large caps<br>Large caps, italics | Standard<br>Standard, italics (left or right slant)<br>Large caps<br>Underscore | Standard<br>Wide<br>Italics |
| Copies | 5 | 4 (ink-jet) | 4 | 1 | 5 |

*) Upon request only. For more information, please refer to the relevant documentation on the TRANSDATA 810 Terminal System.

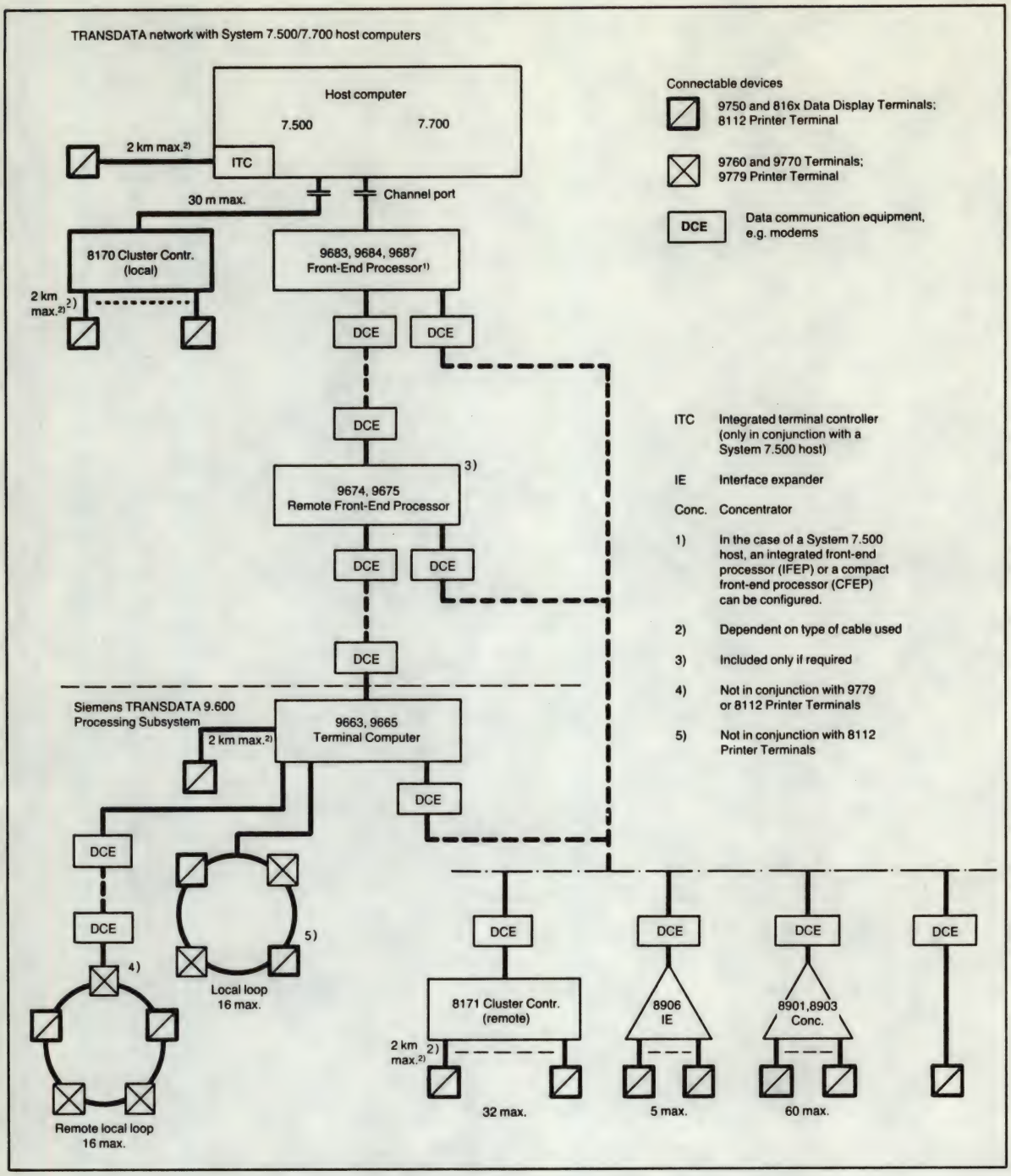
## 1.4 Further connection options (TRANSDATA 960 and System 7.500 configurations)

In TRANSDATA 960 and System 7.500 configurations, the 9750 Data Display Terminal can be operated via the following integrated controllers:

- local controller for
  - communication computers from the TRANSDATA 960 Communication Computer System
  - integrated front-end processors (IFEPs) and compact front-end processors (CFEPs) from System 7.500
- System 7.500 integrated terminal controller (ITC).

Similarly to the cluster controllers, the above-mentioned controllers group together the data traffic of a number of terminals. For more information about operating capabilities and connection options, please refer to the relevant technical documentation available from Siemens AG.

## 1.4.1 Connection and network schematic (TRANSDATA 960 and System 7.500/7.700 configurations)



## 2 Message Transmission

In the case of standalone terminals, message transmision is handled by the synchronous protocol MSV1 or the asynchronous protocol LSV1.

In the case of clustered terminals, message transmission between terminal and cluster controller is handled by the BAM protocol (bit-serial communication protocol for connection to cluster controllers). Transmission between cluster controller and host system is handled by the MSV1 protocol or, for local hookups, via the software interface NEA1.

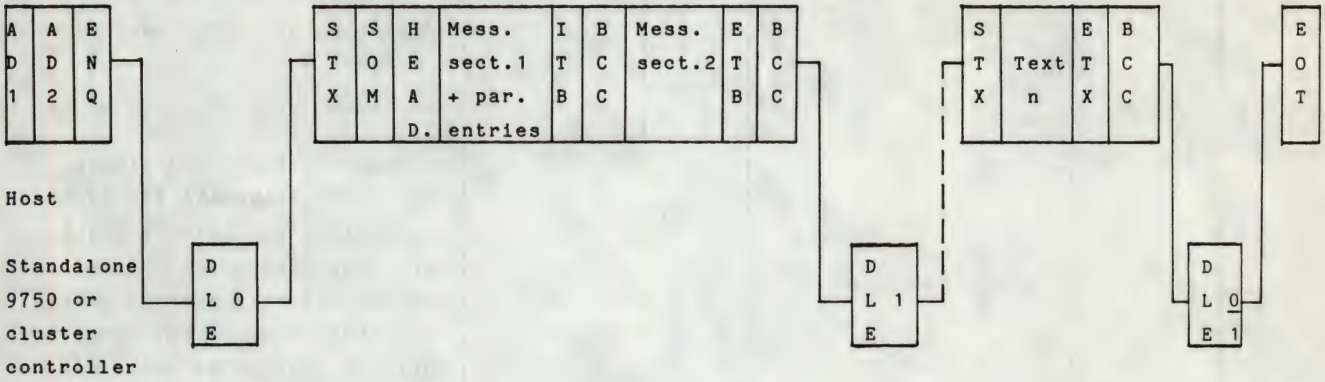
### 2.1 Standalone 9750 Data Display Terminal

#### 2.1.1 Transmission-block format

Before receiving a message, a system component must first be activated by specifying control characters and addresses, i.e. a polling or selection sequence (in accordance with the MSV1 or LSV1 protocol) is sent ahead of the message itself as a separate block. After a positive acknowledgement has been received, the actual message (header plus text sections) can be sent.

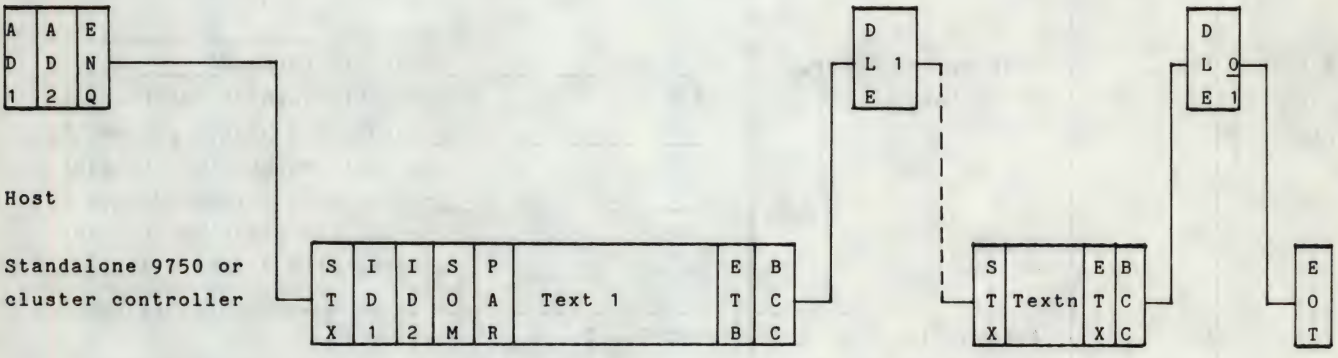
Host to terminal

Selecting




Terminal to host

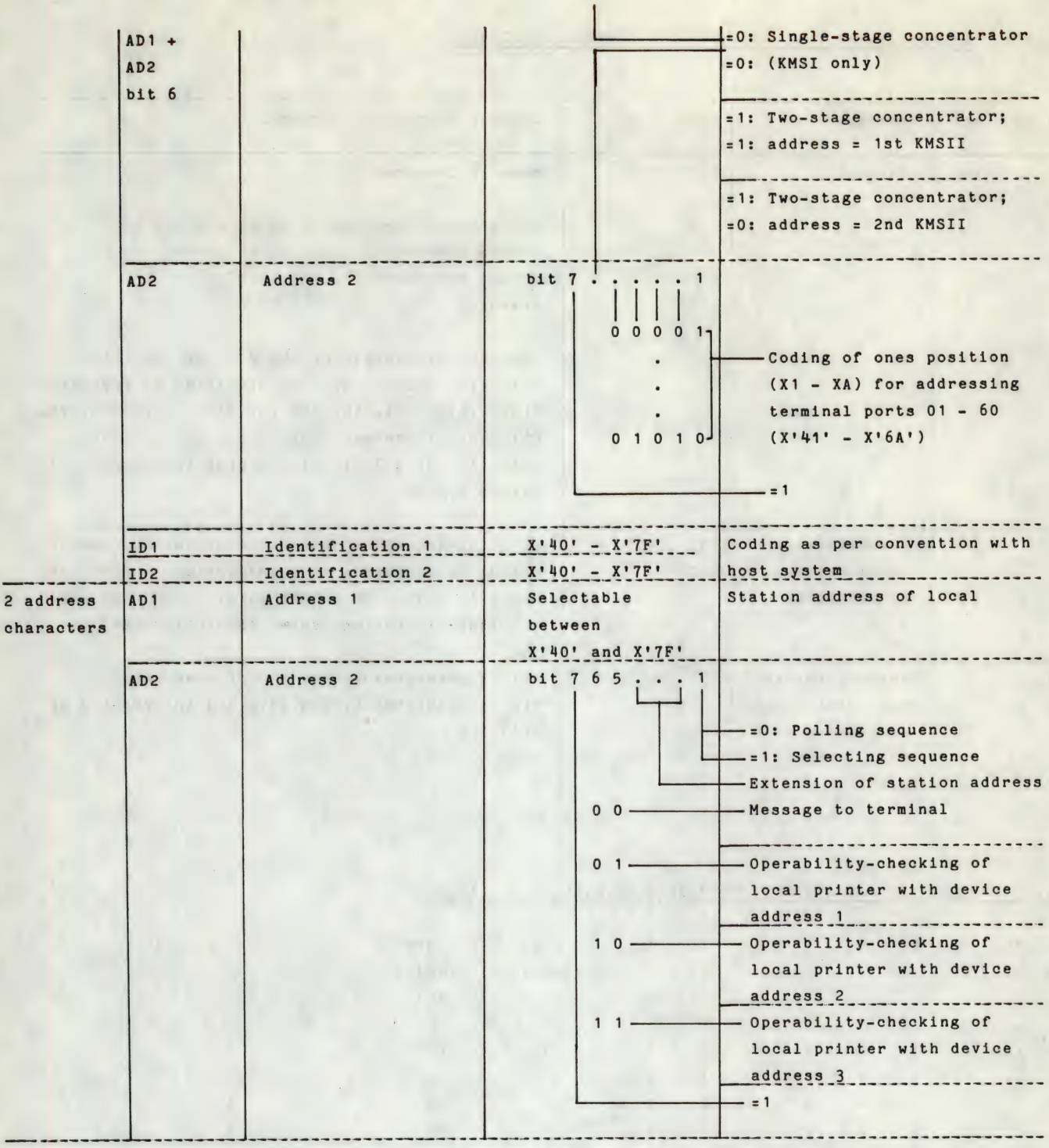
Polling



## 2.1.2 Address and ID formats for standalone terminals

The table below contains the address characters used for addressing standalone terminals. The address character codes are represented alternately in hexadecimal notation or as individual bits.

| Mode | Character | Meaning | Coding (hex notation or bit configuration) | Explanation |
|---|---|---|---|---|
| 1 address character | AD1 | Address 1 | bit 7 6 5 . . . 1 | =0: Polling sequence<br>=1: Selecting sequence<br>Extension of station address |
| | | | 0 0 | Message to terminal |
| | | | 0 1 | Operability-checking of local printer with device address 1 |
| | | | 1 0 | Operability-checking of local printer with device address 2 |
| | | | 1 1 | Operability-checking of local printer with device address 3 |
| | | | =1 | To ensure uniform addressing within the TRANSDATA 810 Terminal System, two address characters should be used. If only one address character is necessary, this must be set accordingly on board ESP. (See adjustment instructions.) |
| Concentrator; poll format = ADR | ADR | Address character, polling | X'70' | |
| Selecting format = AD1/AD2/ ADE | ADE | Address character, selecting | X'41' | Output to terminal |
| | | | X'51' | Operability-checking of local printer with device address 1 |
| | | | X'61' | Operability-checking of local printer with device address 2 |
| | | | X'71' | Operability-checking of local printer with device address 3 |
| | AD1 | Address 1<br>AD1 and AD2 (in concentrator) specify terminal port. | bit 7 . 5 . . . 1 | =1: selecting sequence |
| | | | 1 0 0 0 0 1 | Coding of tens position (1X - 6X) for addressing terminal ports 01 - 60 (X'43' - X'6D') |
| | | | . | |
| | | | . | |
| | | | . | |
| | | | 1 1 0 | |




## 2.2 Clustered 9750 Data Display Terminals

If the configuration contains a local cluster controller and the direction of transmission is to be from host to terminal, the host first sends the address of the terminal required to the controller. This establishes a connection to the terminal, and the message can then be sent. In the case of transmission from terminal to host, the controller recognises a send request from a connected terminal by means of a polling sequence, which is triggered via the DÜ key, and then sends the address of the terminal concerned before taking the message from the terminal character-serially.

## 2.2.1 Addresses and IDs, 8170 Cluster Controller (local)

| | | | |
|---|---|---|---|
| AD1 (8 bits) | Address 1 | X'E0' - X'FF' | Channel address of terminal |
| AD2 (8 bits) | Address 2 | X'CF' | Message to terminal |
| | | X'DF' | Operability-checking of local printer with device address 1 |
| | | X'EF' | device address 2 |
| | | X'FF' | device address 3 |
| | | | The channel address of the 8170 on the byte multiplex channel must be specified in parameter XLINE (LINE=) in the CCP for 8170 operation. The MUX channel address must be set on pc board SA3-K in the 8170 (= line number in communication system). |
| ID1 | Identification 1 | ID1 ≙ AD1 | ID1, ID2 and NKE1/2 are not sent to the communication user program if transmission is from terminal to host. In a CGET call, the CCP enters the logical station number in CALLINFORMATION. |
| | Identification 2 Generated by 8170 | X'CF' | |
| NKA<br>NKE | Host-to-terminal and terminal-to-host message ID bytes respectively | | NKA is generated by the I/O program NEA1.<br>NKE is generated by the 8170 and interpreted by NEA1. |

Coding for channel address AD1, 8170 Cluster Controller (local)

| Bit 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | Line connector | Channel address AD1 | Terminal |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | E0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | E2 | 2 |
| 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | E4 | 3 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | E6 | 4 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 2 | E8 | 5 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 2 | EA | 6 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 2 | EC | 7 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 2 | EE | 8 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 3 | F0 | 9 |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 3 | F2 | 10 |
| 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 3 | F4 | 11 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 3 | F6 | 12 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 4 | F8 | 13 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 4 | FA | 14 |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 4 | FC | 15 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 4 | FE | 16 |

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 5 | E1 | 17 |
| 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 5 | E3 | 18 |
| 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 5 | E5 | 19 |
| 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 5 | E7 | 20 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 6 | E9 | 21 |
| 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 6 | EB | 22 |
| 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 6 | ED | 23 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 6 | EF | 24 |
| 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 7 | F1 | 25 |
| 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 7 | F3 | 26 |
| 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 7 | F5 | 27 |
| 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 7 | F7 | 28 |
| 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 8 | F9 | 29 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 8 | FB | 30 |
| 1 | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 8 | FD | 31 |
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 8 | FF | 32 |

## 2.2.2 Addresses and IDs, 8171 Cluster Controller (remote)

| Character | Meaning | Coding (hex notation or bit configuration) | Explanation |
|---|---|---|---|
| AD1 | Address 1 | X'40'<br>X'60' - X'7F' | General poll, 8171 (polling only)<br>Specific poll, terminal (channel address of 8171-controlled terminal) |
| AD2 | Address 2 | bit 7 . . . . . 1<br>=0: Polling sequence<br>=1: Selecting sequence<br>Station address of 8171 in a multipoint circuit<br>0 0 — Message to terminal<br>0 1 — Operability-checking of local printer with device address 1<br>1 0 — device address 2<br>1 1 — device address 3<br>=1 | |
| ID1 | Identification 1 | X'60' - X'7F' | Channel address of 8171-controlled terminal; generated in 8171 |
| ID2 | Identification 2 | X'40' - X'7F' | ID of 8171 Cluster Controller (remote)<br>- In the case of general polling of the 8171, ID character interpretation must be performed by the host.<br>- In the case of specific polling of a terminal via the 8171, interpretation is optional. |

Coding of channel address AD1, 8171 Cluster Controller (remote)

| Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | Line connector | Channel address AD1 General poll 1) | Channel address AD1 Specific poll 1) | Terminal |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | | 0 | 0 | 0 | 0 | 0 | 1 | 40 | 60 | 1 |
| 1 | | 0 | 0 | 0 | 1 | 0 | 1 | 42 | 62 | 2 |
| 1 | | 0 | 0 | 1 | 0 | 0 | 1 | 44 | 64 | 3 |
| 1 | | 0 | 0 | 1 | 1 | 0 | 1 | 46 | 66 | 4 |
| 1 | | 0 | 1 | 0 | 0 | 0 | 2 | 48 | 68 | 5 |
| 1 | 1) | 0 | 1 | 0 | 1 | 0 | 2 | 4A | 6A | 6 |
| 1 | | 0 | 1 | 1 | 0 | 0 | 2 | 4C | 6C | 7 |
| 1 | | 0 | 1 | 1 | 1 | 0 | 2 | 4E | 6E | 8 |
| 1 | | 1 | 0 | 0 | 0 | 0 | 3 | 50 | 70 | 9 |
| 1 | | 1 | 0 | 0 | 1 | 0 | 3 | 52 | 72 | 10 |
| 1 | | 1 | 0 | 1 | 0 | 0 | 3 | 54 | 74 | 11 |
| 1 | | 1 | 0 | 1 | 1 | 0 | 3 | 56 | 76 | 12 |
| 1 | | 1 | 1 | 0 | 0 | 0 | 4 | 58 | 78 | 13 |
| 1 | | 1 | 1 | 0 | 1 | 0 | 4 | 5A | 7A | 14 |
| 1 | | 1 | 1 | 1 | 0 | 0 | 4 | 5C | 7C | 15 |
| 1 | | 1 | 1 | 1 | 1 | 0 | 4 | 5E | 7E | 16 |
| 1 | | 0 | 0 | 0 | 0 | 1 | 5 | 41 | 61 | 17 |
| 1 | | 0 | 0 | 0 | 1 | 1 | 5 | 43 | 63 | 18 |
| 1 | | 0 | 0 | 1 | 0 | 1 | 5 | 45 | 65 | 19 |
| 1 | | 0 | 0 | 1 | 1 | 1 | 5 | 47 | 67 | 20 |
| 1 | | 0 | 1 | 0 | 0 | 1 | 6 | 49 | 69 | 21 |
| 1 | | 0 | 1 | 0 | 1 | 1 | 6 | 4B | 6B | 22 |
| 1 | | 0 | 1 | 1 | 0 | 1 | 6 | 4D | 6D | 23 |
| 1 | 1) | 0 | 1 | 1 | 1 | 1 | 6 | 4F | 6F | 24 |
| 1 | | 1 | 0 | 0 | 0 | 1 | 7 | 51 | 71 | 25 |
| 1 | | 1 | 0 | 0 | 1 | 1 | 7 | 53 | 73 | 26 |
| 1 | | 1 | 0 | 1 | 0 | 1 | 7 | 55 | 75 | 27 |
| 1 | | 1 | 0 | 1 | 1 | 1 | 7 | 57 | 77 | 28 |
| 1 | | 1 | 1 | 0 | 0 | 1 | 8 | 59 | 79 | 29 |
| 1 | | 1 | 1 | 0 | 1 | 1 | 8 | 5B | 7B | 30 |
| 1 | | 1 | 1 | 1 | 0 | 1 | 8 | 5D | D | 31 |
| 1 | | 1 | 1 | 1 | 1 | 1 | 8 | 5F | 7F | 32 |

1) Bit 6 = 0: General polling of 8171 Cluster Controller (remote) (polling only)

Bit 6 = 1: Specific polling of 8171 Cluster Controller (remote) (polling and selecting)

## 2.3 Message format

A message may comprise the following:

- message header
- text ID
- text blocks.

A message header is always included in every I/O operation, while text IDs are optional and the number of text blocks is not fixed.

### 2.3.1 Message header (host-to-terminal transmission)

In the case of transmission from host to terminal, the message header is represented by the start-of-message (SOM) character. If not followed by a header, the SOM character is always X'40'. (The length of parameter entries (PAG) is not specified in the SOM character.)

### 2.3.2 Message header (terminal-to-host transmission)

In the case of transmission from terminal to host, the message is always preceded by a message header. The header comprises the SOM character and - at least - the eight-byte parameter range PAR 00E. In this case, the SOM character takes on the value X'48'. If a text ID is included, the header comprises the parameter ranges PAR 00E and PAR 01E. In this case, the SOM character takes on the value X'50'.

### 2.3.3 Coding of start-of-message character

If the SOM character is followed by a header (format as per Section 2.5), the length of the header must be specified with the SOM character. Apart from specifying the length of a subsequent message header, the SOM character is also used to identify positive or negative acknowledgements. (Length is not explicitly encoded in the SOM character.)

| Direction of transmission | Coding Hex | Bits (7 . . 4 . . 1) | Meaning |
|---|---|---|---|
| Host to terminal | X'40' | 1 0 0 0 0 0 0 | No message header |
| | X'48' | 1 0 0 1 0 0 0 | Header = PARAM 0 |
| | X'50' | 1 0 1 0 0 0 0 | Header = PARAM 0 + PARAM 1 |
| | X'52' | 1 0 1 1 0 0 0 | Header = PARAM 0 + PARAM 1 + PARAM 2 |
| Terminal to host | X'41' | 1 0 0 0 0 0 1 | Positive acknowledgement |
| | X'42' | 1 0 0 0 0 1 0 | Negative acknowledgement |
| | X'48' | 1 0 0 0 1 0 0 | Transmit with PAR 00E |
| | X'50' | 1 0 1 0 0 0 0 | Transmit with PAR 00E + 01E |
| | X'58' | 1 0 1 1 0 0 0 | Transmit with PAR 00E + 01E + 02E (field mode) |

## 2.4 Parameter ranges

Parameter ranges are specific memory areas within the data display terminal that can be loaded by means of parameter entries, e.g. with regard to executing device functions and send commands, text IDs and operational statuses.

Once loaded, parameter ranges remain valid until they are overwritten by new parameter entries or a message header, or until they are erased via the LSP function (depending on status of character 3 of PAR 001).

NB: If the parameter ranges are loaded by means of a message header, parameter ranges loaded via parameter entries will be erased.

When the data display terminal is powered down, parameter range contents are erased. Upon power-up, default values are automatically loaded into the parameter ranges by the terminal itself.

### 2.4.1 Parameter range overview

| | Parameter ranges for functions initiated by the terminal user via: | | | | | | | | Parameter ranges for functions initiated after end of transmission via an entry in PAR 00D | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Keys | DÜ [1] | LA1 | LA2 | LA3 | LA4 | LA5 | LA6 | LA7 | | LA1 | LA2 | LA3 | LA4 | LA5 | LA6 | LA7 |
| ADDRESS OF PAR. SECTION | ADDRESS OF PARAMETER RANGE | | | | | | | | | | | | | | | |
| | X'40' | X'41' | X'42' | X'43' | X'44' | X'45' | X'46' | X'47' | X'48' | X'49' | X'4A' | X'4B' | X'4C' | X'4D' | X'4E' | X'4F' |
| X'40' | PAR00L | PAR01L | PAR02L | PAR03L | PAR04L | PAR05L | PAR06L | PAR07L | PAR00D | PAR10D | PAR20D | PAR30D | PAR40D | PAR50D | PAR60D | PAR70D |
| X'41' | PAR11L | PAR12L [3] | | | | | | | [2] PAR01D | | | | | | | |
| X'42' | | | | | | | | | [2] PAR02D | | | | | | | |
| X'43' | | | | | | | | | [2] PAR03D | | | | | | | |

| | |
|---|---|
| X'50' | ENTRIES FOR SYSTEM LINE |

For reasons of compatibility, output of a message header causes parts of parameters PARAM 0, PARAM 1 and PARAM 2 to be entered in PAR 00L and 00D, PAR 10L and 10D, and PAR 20L and 20D respectively.

1) All DÜ (Send), F and K keys.
2) Mailbox facility for storing a text ID (entry in PAR 01L).
3) PAR 02L is used in conjunction with field mode only.

## 2.4.2 Parameter range functions

| Parameter range | Function |
|---|---|
| PAR 00L and PAR 01L | These parameter ranges contain values for text manipulation and function control. In the case of terminal-to-host transmission, character FST3 of PAR 00L specifies the length of the message header. |
| PAR 02L | Used for field-related specifications in field mode only. |
| PAR 00D | Used for device functions that are to be executed after transmission has been completed. Host initiation only.<br><br>Character LAP1 may be used for setting bypass mode. Character AZL may be used for activating the system status line. |
| PAR 01D through PAR 03D | Used for text IDs that are included in terminal-to-host transmissions. The length of the text ID is given in character TIL of PAR 01L. |
| PAR 10L through PAR 70L | Used for entering a print command or the channel address of a printer terminal connected to the same cluster controller. Initiation is via the key corresponding to the parameter range, e.g. actuating key LA2 initiates the function entered in parameter PAR 20L. |
| PAR 10D through PAR 70D | Format as for PAR 10L through PAR 70L. LA key function initiated by host. |

## 2.4.3 Loading the parameter ranges with parameter entries

Parameter entries may be included in the text portion of a message. However, for reasons of clarity it is recommended that parameter entries be inserted immediately after the start-of-message character. The SOM character indicates that there is no message header (= X'40').

Parameter entries are identified by a parameter identifier character (PAK). Storing of the parameter entries in the appropriate terminal parameter ranges is controlled by means of the parameter range address and the parameter section address (see Section 2.4.1.), thus permitting exact addressing and modification of every terminal parameter range.

Format of parameter entries

| Character | Code | Meaning |
|---|---|---|
| 1 | X'1B' | Code extension character |
| 2 | X'20' | Identifier character for a three-character sequence |
| 3 | X'61' | Parameter identifier character |
| 4<br>Address of<br>parameter range | X'40' (PAR 00L) | Parameter range for text manipulation and function control |
| | X'41' - X'47'<br>(PAR 10L - PAR 70L) | Parameter range for output to a local printer or to a printer terminal; user-initiated via keys LA1 - LA4 |
| | X'48' (PAR 00D) | Parameter range for host initiation of a device function |
| | X'49' - X'4F'<br>(PAR 10D - PAR70D) | Parameter range for print output; host-initiated via LA function in PAR 00D |
| 5<br>Address of<br>parameter section | X'40' | Base parameter range (PAR 00L/D - PAR 70L/D) |
| | X'41' | First extension, PAR 01L/D |
| | X'42' | Second extension, PAR 02D |
| | X'43' | Third extension, PAR 03D |
| | X'50' | Entries for system status line |
| 1<br>.<br>8<br>1<br>.<br>8<br>1<br>.<br>8<br>1<br>.<br>8 | | Location of actual parameter entries. One parameter range with four eight-byte parameter sections can be loaded with parameter entries. Entries are written continuously to the addressed parameter section of the addressed parameter range until an end character is recognized. |
| | X'21' | End character, which must be located at the end of a parameter entry sequence. If no end character is included, any characters entered after the addressed parameter range has been filled are lost.<br><br>NB: The end character must not be used as the first character of a parameter section. |

## 2.4.4 Operating the 9750 Data Display Terminal on the basis of parameter entries

### Message formatting options

(1) Parameter entries without message, e.g. for parameter loading

(2) Messages without parameter entries, e.g. for simple dialog

(3) Parameter entries plus message, e.g. a message with special commands.

The start-of-message character is coded X'40'. Parameters are identified by means of the parameter entry command X'1B 20 61', each parameter being individually addressable.

### Examples of message formats

```
S
O          n PAR
M

S
O          n PAR          Message
M

S
O          Message        n PAR
M

S
O          n PAR          Message        n PAR
M

S
O          Message        n PAR          Message
M
```

SOM = start-of-message character = X'40'

PAR = parameter entries PAR xyL or PAR xyD

x = 0 - 7
y = 0 - 3

## 2.4.5 Format of parameter range PAR 00L

The parameter range PAR 00L can be loaded by means of parameter entries as well as PARAM0 of the message header. If a message header is used, the content of PARAM0 is moved to the terminal parameter range PAR 00L or PAR 00D. Depending on the content of character 3 of PARAM0 or of PAR 00L, erasing of the parameter range via the keyboard (LSP function) is either possible or not possible.

PAR 00L comprises the following characters:

| Character | Designation | Meaning |
|---|---|---|
| 1 | SAW | Send command |
| 2 | FST1 | Function control character 1 |
| 3 | FST2 | Function control character 2 |
| 4 | GEF1 | Device function character 1 |
| 5 | FST3 | Function control character 3 |
| 6 | FST4 | No meaning assigned |
| 7 | GEF2 | No meaning assigned |
| 8 | WAR | Queue control character |

Format and significance of characters in PAR 00L (host-to-terminal transmission)

| Character | Designation | Coding | Command | Remarks |
|---|---|---|---|---|
| 1 | SAW 1 (Send command 1) | X'40' | Send unprotected fields | The character SAW 1 is interpreted when DÜ1 or F functions are initiated. In the case of transmission from host to terminal, the code of the send command used is moved to PAR 00E.<br><br>Invalid codes are considered as "Send unprotected fields" commands (X'40'). |
| | | X'41' | Send modified fields | |
| | | X'42' | Send unprotected fields from cursor | |
| | | X'43' | Send display buffer contents (including all IS sequences) | |
| | | X'44' | Send unprotected fields, without null characters | |
| | | X'45' | Send modified fields, without null characters | |
| | | X'46' | Send unprotected fields (from cursor), without null characters | |
| | | X'47' | Reserved for field mode (See Appendix 1, Section 1.2.2.) | |

Format and significance of characters in PAR 00L (host-to-terminal transmission) (cont.)

| Character | Designation | Code | Command | Remarks |
|---|---|---|---|---|
| | | X'48' | Send from start marker to cursor, without null characters | Sending from the nearest start marker preceding the cursor |
| | | X'4C' | Send unprotected fields, with relevant null characters | |
| | | X'4D' | Send modified fields, with relevant null characters | |
| | | X'4E' | Send unprotected fields from cursor, with relevant null characters | |
| | | X'4F' | Send only modified fields, without null characters | |

Send data in formatted mode as a function of the status of field handling characters (FBZ) and send commands (SAW)

| SAW code | | | $48_{(16)}$ | $40_{(16)}$ | $44_{(16)}$ | $4C_{(16)}$ | $42_{(16)}$ | $46_{(16)}$ | $4E_{(16)}$ | $41_{(16)}$ | $45_{(16)}$ | $4D_{(16)}$ | $4F_{(16)}$ | $43_{(16)}$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | Send unprotected fields | | | | | | | Send modified fields | | | | Send display buffer contents |
| | | | from start marker or start of screen | from start of buffer | | | from cursor | | | from start of buffer | | | | from start |
| Bit set in FBZ | | | to cursor | to end marker or end of buffer | | | | | | | | | | to end 2) |
| 1 | 6 | 3 | with relevant NULs | with NULs | without NULs | with relevant NULs | with NULs | without NULs | with relevant NULs | with NULs | without NULs | with relevant NULs | without NULs | with NULs |
| 0 | 0 | 0 | Address of start marker or start of screen + start marker + field address + field contents 1) | Field contents | Field addresses + field contents | | Cursor address + field contents | Cursor addr. + field addr. + field contents | | | | | | |
| | | 1 | | | | | | | | | | | Field addr. + field contents | |
| | 1 | 0 | | | | | | | | Field addresses + field contents | | | | |
| | | 1 | | | | | | | | | | | Field addr. + field contents | |
| 1 | 0 | 0 | | | | | | | | | | | | Text + field separators |
| | | 1 | | | | | | | | | | | Field addr. + field contents | |
| | 1 | 0 | | | | | | | | Field addresses | | | | |
| | | 1 | | | | | | | | | | | Field addr. + field contents | |

1) An end marker is considered a data character.
2) No restrictions.

- EM: Only end markers in send fields are interpreted.
- Relevant null characters: Null characters are relevant null characters if they identify an interval, i.e. if at least one send character or end marker is included in the same field.
- LZE: Characters between a logical end-of-line (LZE) marker and the end of a field or the physical end of a line are not sent.

Format and significance of characters in PAR OOL (host-to-terminal transmission) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 2 | FST 1 | Function control character | bit 7 . 5 . 3 . 1 | Functions EFZ (Insert Line) and AFZ (Delete Line) via keyboard | EFZ and AFZ functions |
| | | | bit 1 | =0: enabled | |
| | | | 1 bit 1 | =1: disabled | |
| | | | bit 2 | =0: Field separators and protected data are copied from last line to cursor line. Unprotected data are replaced by null characters. | EFZ function |
| | | | bit 2 | =1 Field separators and all data are copied from last line to cursor line. | |
| | | | bit 3 | =0 Field separators and protected data are copied from cursor line to last line. Unprotected data are replaced by null characters. | AFZ function |
| | | | bit 3 | =1 Field separators and all data are copied from cursor line to last line. | |
| | | | bit 4 | =0 Cursor function (as on 8161): cursor cannot be moved to protected fields | Different cursor handling if ATAB (autotab) is on |
| | | | bit 4 | =1 Cursor function: cursor can be moved to protected fields. | |
| | | | bit 5 | =0 RU (Rollup) function enabled | RU and RUB functions |
| | | | bit 5 | =1 RU (Rollup) function disabled | |
| | | | bit 6 | =0 RUB (Rollup) mode off | |
| | | | bit 6 | =1 RUB (Rollup) mode on | |

Format and significance of characters in PAR 00L (host-to-terminal transmission) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 3 | FST 2 | Function control character | bit 7 . 5 . 3 . 1 | LSP (Erase Memory) function | |
| | | | bit 1 | =0: keyboard init. enabled | |
| | | | bit 1 | =1: keyboard init. disabled | |
| | | | bit 2 | =0: Param. range PAR 00L - 70L not erased when LSP function initiated | PAR 00L always erased; PAR 01L cannot be erased |
| | | | bit 2 | =1: Param. range PAR 00L - 70L erased when LSP function initiated | |
| | | | bit 3 | =0: Autotab (ATAB) off | |
| | | | bit 3 | =1: Autotab (ATAB) on | |
| | | | bit 4 | =0: FAZ function (field sep. to defined state) initiated after every output | |
| | | | bit 4 | =1: FAZ function initiated only when FAZ is output in message | |
| | | | bit 5 | See table below | |
| | | | bit 6 | =0: If LVD (Erase variable data) is entered, modify and flash bits (MOD and BLINKEN) in unprotected fields are erased between cursor and end marker. | Reaction to bits 5 and 6 being 0 corresponds to reaction with 8161 |
| | | | | =1: If LVD is entered, modify and flash bits in unprotected fields are erased between start of screen and end of screen. | |
| | | | bit 7 | =1: For avoiding transmission control characters. | |

Format and significance of characters in PAR 00L (host-to-terminal transmission) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|

Manipulation of modify and flash bits (MOD and BLINKEN) in conjunction with the function LZF (Erase to end of line or end of field), depending on status of char. FST 2, bit 5

| from | to | Mod. bit Char. FST 2, bit 5 = 0 | Mod. bit Char. FST 2, bit 5 = 1 | Flash bits Char. FST 2, bit 5 = 0 | Flash bits Char. FST 2, bit 5 = 1 |
|---|---|---|---|---|---|
| Start of field | End of field | ↓ | ↑ | ↓ | - |
| Start of field | Any pos. in field | ↓ | ↑ | ↓ | |
| Any pos. in field | Any pos. in field | - | ↑ | - | - |
| Any pos. in field | End of field | - | ↑ | - | - |

↑ Set

↓ Cleared

\- Unmodified

Format and significance of characters in PAR 00L (host-to-terminal transmission) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 4 | GEF 1 | Device function char. 1 | See sections on device functions. | Character 2 (device function) is moved to char. 1 (GEF 1) of parameter range PAR 00D (only for terminal-to-host transm. via message header) | Char. 4 is interpreted only for terminal-to-host transm. (via message header). |
| 5 | FST 3 | Function control char. 3 | bit 7 . 5 . 3 . 1 | | |
| | | | bit 1 =0: | Cursor pos. tracks during host-to-terminal transm. | Cursor pos. control for host-to-terminal trans. |
| | | | bit 1 =1 | Cursor pos. controlled via device function SS (Set cursor) | |
| | | | bit 2 =0 | For terminal-to-host transm., positioning commands and cursor pos. are specified in PAR01E in 8161-compatible code (Table 3.2.1) | Cursor positioning. For coding, see Section 3.2. |
| | | | bit 2 =1 | For terminal-to-host transm., positioning commands and cursor pos. are specified in PAR01E as per coding in Table 3.2.2. | |
| | | | bit 3 =0 | For terminal-to-host transm., the message header comprises PAR00E only. | Controls PARE function. |
| | | | bit 3 =1 | For terminal-to-host transm., the message header comprises PAR00E plus PAR01E. | |
| | | | bit 4 =0 | After terminal-to-host transm., the cursor is behind the valid end marker or at start of screen. | Cursor pos. control |

Format and significance of characters in PAR 00L (host-to-terminal transmission) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| | | | bit 4 | =1 Cursor pos. not affected by terminal-to-host transm. | |
| | | | bit 5 | =0 Dialog mode not activated. | Dialog mode control |
| | | | bit 5 | =1 Dialog mode activated. After initiation of a transm. function via keyboard, the keyboard is blocked. Transm. from host that involves the display buffer unblocks the keyboard. | |
| | | | bit 6 | =0 Keyboard released. | Short messages to host are always possible. |
| | | | bit 6 | =1 Keyboard blocked. | |
| 6 | FST 4 | Function control char. 4 | | No meaning assigned. | |
| 7 | GEF 2 | Device function 2 | | No meaning assigned. | |
| 8 | WAR | Queue control char. | Any coding; not a transm. control char. | | Can be used for ordering texts received for processing and, during the next transm. to host, is sent unmodified to host (in PAR00E). |

## 2.4.6 Parameter range PAR 01L

The parameter range PAR 01L is used as an extension to the base parameter range PAR 00L and is loadable only by means of parameter entries.

PAR 01L comprises the following characters:

| Character | Designation | Meaning |
|---|---|---|
| 1 | TIL | Text ID length |
| 2 | TKL | Reserved |
| 3 | FST5 | Causes LA keys to be blocked |
| 4 | FST6 | Causes P keys to be blocked |
| 5 | FST7 | Programming capability for P keys disabled |
| 6 | SAW2 | Send command SAW2 |
| 7 | FST8 | Request for badge information entry |
| 8 | * | Reserved |

Format and significance of characters in PAR 01L (host-to-terminal transmission)

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 1 | TIL | Text ID length indicates whether and to what extent a TI is to be sent from the mailbox facilities (BK) to host during next transm. to host (BK = PAR 01D - 03D). | X'00'<br>X'41<br>X'42'<br>X'43' | No text ID (TI)<br>TI = 8 bytes<br>TI = 16 bytes<br>TI = 24 bytes | TI is stored in PAR 01D,PAR 02D and 03D. |
| 2 | TKL | Header length | X'00' | No meaning assigned | |

Format and significance of characters in PAR 01L (host-to-terminal transmission) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 3 | FST5 | Function control char. 5; LA keys blocked for user | bit 7 . 5 . 3 . 1 = 1 | bit 1: 0: LA1 key; 1: | 0= LA function permissible<br>1= LA function not permissible<br>LA functions within P functions are blocked too |
| | | | | bit 2: 0: LA2 key; 1: | |
| | | | | bit 3: 0: LA3 key; 1: | |
| | | | | bit 4: 0: LA4 key; 1: | |
| | | | | bit 5: 0: LA5 key; 1: | |
| | | | | bit 6: 0: LA6 and LA7 keys; 1: | |
| 4 | FST6 | Function control char. 6; P keys blocked for user | bit 7 . 5 . 3 . 1 = 1 | bit 1: 0: P1 key; 1: | 0= LA function permissible<br>1= LA function not permissible |
| | | | | bit 2: 0: P2 key; 1: | |
| | | | | bit 3: 0: P3 key; 1: | |
| | | | | bit 4: 0: P4 key; 1: | |
| | | | | bit 5: 0: P5 key; 1: | |

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| | | | bit 6 — 0: | Keys P6 - P20 | |
| | | | bit 6 — 1: | | |
| 5 | FST7 | Function control char. 7 | 7 . 5 . 3 . 1 | | 0= P key programming permissible |
| | | | 1 — bit 1 — 0: | P1 key | |
| | | | bit 1 — 1: | | 1= P key programming not permissible |
| | | | bit 2 — 0: | P2 key | |
| | | | bit 2 — 1: | | |
| | | | bit 3 — 0: | P3 key | |
| | | | bit 3 — 1: | | |
| | | | bit 4 — 0: | P4 key | |
| | | | bit 4 — 1: | | |
| | | | bit 5 — 0: | P5 key | |
| | | | bit 5 — 1: | | |
| | | | bit 6 — 0: | Keys P6 - P20 | |
| | | | bit 6 — 1: | | |
| 6 | SAW2 | Send command 2 (for DÜ2 only) | X'40' | Send unprotected fields | SAW1 and SAW2 have same function. |
| | | | X'41' | Send modified fields | |
| | | | X'42' | Send from cursor | Invalid codes are treated as "Send unprotected fields" (X'40') |
| | | | X'43' | Send display buffer contents | |
| | | | X'44' | Send unprotected fields, without null chars. | |

Format and significance of characters in PAR 01L (host-to-terminal transmission) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| | | | X'45' | Send modified fields, without null chars. | |
| | | | X'46' | Send from cursor, without null chars. | |
| | | | X'48' | Send from start marker to cursor without null chars. | The valid start marker is the one immediately preceding the cursor. |
| | | | X'4C' | Send unprotected fields with relevant null chars. | |
| | | | X'4D' | Send modified fields with relevant null chars. | |
| | | | X'4E' | Send unprotected fields from cursor, with relevant null chars. | |
| | | | X'4F' | Send modified fields only, without null chars. | |
| 7 | FST8 | Function control char. 8; badge information request | bit<br>7 . 5 . 3 . 1<br>1 0 0 0 0 0 | 0: Badge entry blocked<br>1: Badge entry expected (keyboard entry blocked) | |
| 8 | | | | Reserved | |

PAR 02L is reserved for field mode. (See Appendix 1 for description.)

## 2.4.7 Parameter range PAR 00D

Terminal parameter range PAR 00D, which is downline-loaded from the host, is reserved for device functions initiated after completion of host-to-terminal transmission. The parameter range can be loaded via special parameter entries or via PARAM0 of a message header. If a message header is used, the values relating to device function implementation are automatically moved from PAR 00L -GEF1 to PAR 00D.

PAR 00D comprises the following characters:

| Character | Designation | Meaning |
|---|---|---|
| 1 | GEF1 | Device function character 1 |
| 2 | GEF2 | Device function character 2 |
| 3 | | Reserved |
| 4 | LAP1 | LA function parameter 1 |
| 5 | AZL | Status line |
| 6 | | Reserved |
| 7 | | Reserved |
| 8 | | Reserved |

Format and significance of characters in PAR 00D (host-to-terminal transmission)

| Character | Desig. | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 1 | GEF1 | Device function char. 1 | As per device function code table<br><br>(Two-char./ three-char. sequence) | The device function entered is initiated after termination of host-to-terminal The following may be entered:<br>1. 2nd char. of a two-char. sequence (with initiation of corresponding device function);<br>2. Identifier character of a three-char. sequence (X'20') (with initiation of corresponding device function by GEF2). | The code extension char. ESC must not be entered. GEF X'00' does not cause initiation of a device function. Device function is implemented as for keyboard initiation. |

Format and significance of characters in PAR 00D (host-to-terminal transmission) (cont.)

| Character | Desig. | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 2 | GEF2 | Device function char. 2 | Corresponds to 3rd char. of a three-char. seq. | The device function entered is initiated after termination of host-to-terminal transmission. | Only interpreted if GEF1 contains the identifier character of a three-character sequence (SP = X'20') |
| 3 | | | | Reserved | |
| 4 | LAP1 | LA function parameter | X'40' | Transmission to and from the host is handled via the terminal screen. Device functions within a message are implemented. | |
| | | | X'41' | Output to local devices (printers) is handled without modifying screen contents. See bypass mode. | |
| 5 | AZL | Status line | X'40' | Operating status displayed. | |
| | | | X'41' | System line displayed. | |
| 6<br>7<br>8 | | | X'00' | No meaning assigned. | |

## 2.4.8 Parameter entries in PAR 10L/D through PAR 70L/D

These parameters refer to intra-system data interchange. (See Section 5.)

## 2.4.9 Functions and format of parameter ranges PAR 00E through PAR 02E (terminal-to-host transmission)

For each transmission to the host, the terminal precedes the message with parameter range PAR 00E or PAR 01E, depending on the status of FST3 of PAR 00L. In field mode, PAR 02E is also included. (See Appendix 1 for description.) If a message header is used, only PAR 00E is sent.

Parameter ranges for terminal-to-host transmission

| Parameter range | Function | Remarks |
|---|---|---|
| PAR 00E | Entering device-specific information such as buffer length, line length, revision level etc. | The information is taken from PAR 00L or PAR 01L. For every terminal-to-host transmission, it is regenerated and sent to the host as a message header. The length of the header is given in the SOM (start-of-message) character. |
| PAR 01E | Extension of PAR 00E, for entering text ID length and header length and for specifying the cursor position before initiation of transmission. | |
| PAR 02E | Reserved for field mode. | See Appendix 1 for description. |

## 2.4.10 Parameter range PAR 00E

Parameter range PAR 00E comprises the following characters:

| Character | Designation | Meaning |
|---|---|---|
| 1 | SAW | Send command used for terminal-to-host transmission |
| 2 | PFLNG | Buffer length |
| 3 | ZLNG | Line length |
| 4 | AUSG | Program revision level |
| 5 | ZZ1 | Status character 1 |
| 6 | ZZ2 | Status character 2 |
| 7 | CDS | Send-key code |
| 8 | WAR | Queue control character |

Format and significance of characters in PAR 00E

| Char. | Desig. | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 1 | SAW | Send command | Dependent on whether SAW1 or SAW2 is used. | Depending on the transmit key used (DÜ1 or DÜ2), the corresponding send command (SAW1 or SAW2) is entered in PAR 00E and executed.<br><br>The functions SAW1 and SAW2 are specified (via host-to-terminal transm.) in PAR 00L and PAR 01L respectively. | After terminal power-up or user initiation of LSP function, SAW is X'00'. |
| 2 | PFLNG | Buffer length | X'40' | Set buffer length: 4K characters | Buffer length and line length are preset.Characters 2 and 3 should not be interpreted. |
| 3 | ZLNG | Line length | X'30' | Line length: 80 chars. | |
| 4 | AUSG | Program revision level | X'60' | The host is informed of the terminal's program revision level. | Program memory DIALOG (-V101, GS4), or DIALOG + BERMUDA (-V201, GS2) |
| | | | X'67' | | Program memory DIALOG (-V101, GS5) |
| | | | X'64' | | Program memory DIALOG (-V303, GS1); P keys resident ** |
| | | | X'65' | | Program memory DIALOG + BERMUDA with keyboard crosschecking; P keys resident (-V202, GS1) ** |
| | | | X'66' | | Program memory DIALOG for X.21; P keys resident ** |

** When the device is powered down, the functions programmed for keys P1 - P5 are retained.

| Char. | Desig. | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| 5 | ZZ1 | Status char. 1 is an extension of ZZ2 and contains the user ID | bit 7 . 5 . 3 . 1 | | bit x = 0: OFF (AUS)<br>bit x = 1: ON (EIN) |
| | | | 1 — bit 1 | =0: No badge reader ID | |
| | | | 1 — bit 1 | =1: Badge reader ID present (badge inserted) | |
| | | | 1 — bit 2 | =0: Keylock switch 1 | |
| | | | 1 — bit 2 | =1: | |
| | | | bit 3 | =0: Keylock switch 2 | |
| | | | bit 3 | =1: | |
| | | | bit 4 | =0: Keylock switch 3 | |
| | | | bit 4 | =1: | |
| | | | bit 5 | =0: Local device with GAD6 | bit x = 0: Device inoperable, i.e. disconnected or malfunctioning (device-specific status message) |
| | | | bit 5 | =1: | |
| | | | bit 6 | =0: Local device with GAD7 | |
| | | | | =1 | bit x = 1: Device operable |
| 6 | ZZ2 | Status char. 2, for operability of local devices (incl. keyboard) | bit 7 . 5 . 3 . 1 | | bit x = 0: Device inoperable, i.e. disconnected or malfunctioning (device-specific status message)<br>bit x = 1: Device operable |
| | | | 1 — bit 1 | =0: Keyboard | |
| | | | 1 — bit 1 | =1: | |
| | | | 1 — bit 2 | =0: Local device with GAD1 | |
| | | | 1 — bit 2 | =1: | |
| | | | bit 3 | =0: Local device with GAD2 | |
| | | | bit 3 | =1: | |

Format and significance of characters in PAR 00E (cont.)

| Char. | Desig. | Meaning | Coding | Command | Remarks |
|---|---|---|---|---|---|
| | | | bit 4 =0: Local device with GAD3<br>bit 4 =1:<br>bit 5 =0: Local device with GAD4<br>bit 5 =1:<br>bit 6 =0: Local device with GAD5<br>bit 6 =1: | | |
| 7 | CDS | Send-key code (K, F or DÜ keys, with confirmation as appropriate) | See table of transmit functions. | | Function as per table of transm. functions. |
| 8 | WAR | Queue control character | Any coding poss.; not a comm. control char. | | The terminal returns the host-specified WAR to the host unmodified. |

## 2.4.11 Parameter range PAR 01E

Parameter range PAR 01E comprises the following characters:

| Character | Designation | Meaning |
|---|---|---|
| 1 | TIL | Text ID length |
| 2 | TKL | Reserved |
| 3 | ZLA | Line address |
| 4 | SPA | Column address |
| 5 | SAD | Page address |
| 6 | FEM | Error messages |
| 7 | * | Reserved |
| 8 | * | Reserved |

Format and significance of characters in PAR 01E

| Char. | Desig. | Meaning | Coding | Command |
|---|---|---|---|---|
| 1 | TIL | Text ID length | X'40'<br>X'41'<br>X'42'<br>X'43' | No text identification (TI)<br>TI = 8 bytes<br>TI = 16 bytes<br>TI = 24 bytes |
| 2 | TKL | Header length | X'00' | Reserved |
| 3 | ZLA | Line address | Dependent on status of FST3, bit 2. See table, 3.2.2. | Characters 3, 4 and 5 indicate cursor position before actuation of DÜ, K or F key. |
| 4 | SPA | Column address | | |
| 5 | SAD | Page address | X'30' | |
| 6 | FEM | Error message | See table, 2.4.14. | For error messages. |
| 7 | * | | X'00' | Reserved |
| 8 | * | | X'00' | Reserved |

See Appendix 1 for a description of the format and significance of characters in PAR 02E.

## 2.4.12 Text ID (TI)

A text ID is a parameter entry that may be freely structured by the user, permitting the identified text portion to be assigned in a particular way. The text ID length (TIL) indicates both whether a text ID is to be sent from the terminal's mailbox to the host and the length of the text ID. If a text ID is stored in the terminal's mailbox (PAR 01D, 02D, 03D), it is sent to the host - every time transmission to the host takes place - immediately behind PAR 01E, depending on the status of character 1 of PAR 01L. (See Section 2.4.6.)

## 2.4.13 System messages

When a message is output from host to terminal, the 25th line of the screen can display a system line instead of the status line. The system line is used for operator prompting and may contain up to 80 visible and 48 invisible display control characters (ASZ). Entering successive display control characters is not permissible because of the risk of errors.

The end marker (EM) is the non-displayed system-message terminating character. After the end marker has been recognized, the rest of the system line is erased down to null characters. IS2 and IS4 sequences are not interpreted in the system line.

If, during execution of a function, an error occurs or P, PAR or EFG mode is activated, the system line is replaced on screen by the status line. The system line is displayed when the error condition is reset via RS or when the above-mentioned modes have been deactivated.

NB: With regard to displaying the system line, it is not practical to include an audible alarm (BELL) or a visual alarm (AKA) because the system line is replaced by the status line, in which the the audible alarm is visually represented.

Parameter range addressing for the system line

| Character | Coding | Command | Meaning |
|---|---|---|---|
| 1 | X'1B' | | |
| 2 | X'20' | | Parameter identifier character (PAK) |
| 3 | X'61' | | |
| 4 | X'40' | Parameter range | Address of system line |
| 5 | X'50' | Parameter section | |
| 6<br>.<br>.<br>134 max. | | Enter system message in parameter range addressed. | Contents of system line, comprising up to 80 displayable and 48 non-displayable control characters. |
| | X'19' | Terminating character for system line | Terminator |

## 2.4.14 Error messages in parameter ranges

The terminal records error messages in character 6 (FEM) of PAR 01E, i.e. instances of non-adherence to conventions, such as

- errored message(s) or
- invalid function(s),

are entered in character FEM, which is sent to the host during the next transmission.

### Error codes in FEM byte of PAR 01E

For an explanation of the function abbreviations, please refer to Sections 3.5.2 through 3.5.8.

| Code | Function | Remarks |
|---|---|---|
| X'40' | MAR | Non-markable field |
| X'41' | WDH | Invalid address |
| X'42' | WDH | Non-null characters preceding logical end of line |
| X'44' | LVA | Invalid address |
| X'46' | PAR | Range full |
| X'47' | PAR | Invalid characters |
| X'48' | PAR | Invalid characters |
| X'4A' | P | Range full (8 bytes max.) |
| X'4B' | P | Invalid character |
| X'4C' | P | Invalid function |
| X'4F' | PAR 00D | Printer control character (ESC sequence) in PAR 00D |
| X'50' | PAK | More than 48 display control characters in system line |
| X'51' | - | ESC sequence invalid or incomplete |
| X'52' | PAK | Parameter entries invalid or incorrect |
| X'53' | FTZ | IS2 or IS3 sequence invalid at this location within the message |
| X'54' | Pos. | Invalid positioning information |

Error codes in FEM byte of PAR 01E (cont.)

| Code | Function | Meaning |
|---|---|---|
| X'55' | DÜ | Invalid function in message |
| X'56' | PAR 00D | Invalid function in PAR 00D |
| X'57' | LA | Invalid function in message |
| X'5A' | SS | Invalid function in PAR 00D |
| X'60' | Host-to-terminal transm. | Characters preceding logical end of line |
| X'61' | EFG | Character entered in protected field |
| X'62' | AFG | Character deleted in protected field preceding logical end of line |
| X'63' | LZF | Invalid character in protected field |
| X'64' | LVD | Invalid character in protected field at end of screen |
| X'65' | EFG | Alphabetic character entered in numeric field |
| X'66' | RUB | Roll-up function disabled in RUB mode |
| X'67' | Host-to-terminal transm. | Keyboard crosschecking invalid in protected field |
| X'6A' | TES | Invalid entry |
| X'6B' | AUS/RUF | X.21 functions from host invalid |
| X'71' | LA | Incorrect printer controller parameters |
| X'75' | LA | Incorrect command in parameter range |
| X'79' | LA | No text in bypass mode |
| X'7E' | Hardware | Hardware error |
| X'7F' | | Acknowledgement buffer overflow |

## 2.5 Message format including header

The 9750 Data Display Terminal can also be addressed by means of a message header and in this the 9750 reacts in the sane way as the 8161 Data Display Terminal.

The parameters are designated PARAM0, PARAM1 and PARAM2; their format corresponds to that of the parameter ranges PAR 00L, PAR 10L and PAR 20L.

For host-to-terminal transmission, the contents of PARAM0, PARAM1 and PARAM2 are entered in parameter ranges PAR 00L/D, PAR 10L/D and PAR 20L/D.

If a message header is output, it is not possible to address a particular parameter range. The length of the message header is contained in the start-of-message character. (See Section 2.3.3 for coding.)

### 2.5.1 Functions of parameters PARAM0 through PARAM2

| PARAM0 | PARAM1 and PARAM2 |
|---|---|
| PARAM0 is used exclusively for communication between the data display terminal and the host and contains device-specific information on how the subsequent text is to be handled. PARAM0 is also included - as parameter range PAR 00E - in terminal-to-host transmission. | PARAM1 and PARAM2 are identical in format and contain information on communication between the data display terminal and a local printer or the printer terminals connected to the cluster controller. In addition, this permits control information to be sent directly from the host to the printer terminal. |

### 2.5.2 Parameter format

The format of PARAM0 corresponds to that of PAR 00L (see Section 2.4.5), while the format of PARAM1 and PARAM2 corresponds to that of PAR 10L and PAR 20L (see Section 5.3.4).

### 2.5.3 Possible error sources

- If a message header is output, the contents of all parameter ranges loaded via parameter entries are cleared and replaced by the contents of the message header.
- If intra-system data interchange is initiated with a message header in PAR 001 (PARAM0), GEF1, a parameter entry for an LA1 function, PAR 10L (PARAM1) must not contain an acknowledgement request.
- If a message header is used, only two-character sequences may be entered in character GEF of PARAM0 (char. GEF1 of PAR 00D).
- In the case of terminal-to-host transmission, the queue control character (WAR) is always ASCII-encoded, i.e. bit combinations > X'7F' are falsified and bit $2^7$ is set to 0.

## 3 Message section format

Message sections can be separated by

- absolute positioning commands (IS4 sequences),
- field separators (FTZ; IS2/IS3 sequences), and
- device functions (ESC sequences in text or entries in PAR 00L/00D).

### 3.1 Message format including positioning commands (IS4 sequences)

Any character position, including preceding field separators, can be addressed by means of positioning commands within a text block, thus permitting modification at any time. The text may contain any number of positioning command sequences in any order. These sequences are identified by the code extension character IS4 (X'1C').

Typical format of a positioning command sequence

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| AA ... AA | IS4 | ZLA | SPA | SAD | ITB ** | B ... B | |
| Text | Positioning command | | | | | Text | |

** Optional

The positioning command comprises:

| | | |
|---|---|---|
| the identifier character | | IS4 (X'1C') |
| the base address | for line | ZLA (X'20' ... '37' or '68' ... '7F') |
| | for column | SPA (X'30' ... '7F') |
| the page address | | SAD (X'30') |
| the (optional) section character | | ITB (X'1F') |

Base address

The base address (line and column) indicates the position of the cursor on screen. Field separators are located before the base address.

Page address

The page address is X'30' on the 9750 Data Display Terminal.

Section character

For device functions whose execution takes the data display terminal some time, such as positioning, it is recommended to subdivide the text with section characters (ITB). After an ITB has been recognized, device functions are implemented immediately if switch S4-1 is set to 0 (settings memory). If

ITBs are not included, functions are not implemented until the text has been terminated by an ETB or ETX character. If switch S4-1 is set to 1, processing does not start until an ETX character has been recognized.

Example of a positioning command

In a host-to-terminal transmission sequence, the cursor is to be positioned in line 3, column 10.

Sequence from host: IS4/ZLA=3/SPA=10/SAD=0/ITB/ ...

## 3.2 Coding of line and column addresses for IS4, WDH and LVA sequences

In the case of host-to-terminal transmission, there are no differences in the coding of line addresses. (See tables in Sections 3.2.1 and 3.2.2.)

In the case of terminal-to-host transmission, both the coding of the line address and the cursor position in PAR 01E are determined by the status of bit 2 of character FST3.

When WDH and LVA sequences are to be output, the line address must be encoded in accordance with the table in Section 3.2.2.

## 3.2.1 8161-compatible coding

**Terminal-to-host transmission, PAR OOL, character 5 (FST3), bit 2 = 0**
**Line address: X'37' - X'20' Column address: X'7F' - X'30'**

| Bits b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Line \ Col. | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| | | | | | | | | | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| | | | | | | | | | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Parity bit | | | | 0 | 0 | 0 | 0 | 0 | | | 24 | 8 80 | 64 | 48 | 32 | 16 |
| | | | | 0 | 0 | 0 | 1 | 1 | | | 23 | 7 79 | 63 | 47 | 31 | 15 |
| | | | | 0 | 0 | 1 | 0 | 2 | | | 22 | 6 78 | 62 | 46 | 30 | 14 |
| | | | | 0 | 0 | 1 | 1 | 3 | | | 21 | 5 77 | 61 | 45 | 29 | 13 |
| | | | | 0 | 1 | 0 | 0 | 4 | | | 20 | 4 76 | 60 | 44 | 28 | 12 |
| | | | | 0 | 1 | 0 | 1 | 5 | | | 19 | 3 75 | 59 | 43 | 27 | 11 |
| | | | | 0 | 1 | 1 | 0 | 6 | | | 18 | 2 74 | 58 | 42 | 26 | 10 |
| | | | | 0 | 1 | 1 | 1 | 7 | | | 17 | 1 73 | 57 | 41 | 25 | 9 |
| | | | | 1 | 0 | 0 | 0 | 8 | | | 16 | 72 | 56 | 40 | 24 | 8 |
| | | | | 1 | 0 | 0 | 1 | 9 | | | 15 | 71 | 55 | 39 | 23 | 7 |
| | | | | 1 | 0 | 1 | 0 | 10 | | | 14 | 70 | 54 | 38 | 22 | 6 |
| | | | | 1 | 0 | 1 | 1 | 11 | | | 13 | 69 | 53 | 37 | 21 | 5 |
| | | | | 1 | 1 | 0 | 0 | 12 | | | 12 | 68 | 52 | 36 | 20 | 4 |
| | | | | 1 | 1 | 0 | 1 | 13 | | | 11 | 67 | 51 | 35 | 19 | 3 |
| | | | | 1 | 1 | 1 | 0 | 14 | | | 10 | 66 | 50 | 34 | 18 | 2 |
| | | | | 1 | 1 | 1 | 1 | 15 | | | 9 | 65 | 49 | 33 | 17 | 1 |

**Coding of line and column address for absolute positioning commands:**
**IS4, ZLA, SPA, SAD (SAD = X'30')**

◪ Line addresses

### 3.2.2 New coding

Terminal-to-host transmission, PAR OOL, character 5 (FST3), bit 2 = 1
Line address: X'7F' - X'68' Column address: X'7F' - X'30'

| Bits b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Line \ Col. | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| | | | | | | | | | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Parity bit | | | | 0 | 0 | 0 | 0 | 0 | | | | 80 | 64 | 48 | 32 | 16 |
| | | | | 0 | 0 | 0 | 1 | 1 | | | | 79 | 63 | 47 | 31 | 15 |
| | | | | 0 | 0 | 1 | 0 | 2 | | | | 78 | 62 | 46 | 30 | 14 |
| | | | | 0 | 0 | 1 | 1 | 3 | | | | 77 | 61 | 45 | 29 | 13 |
| | | | | 0 | 1 | 0 | 0 | 4 | | | | 76 | 60 | 44 | 28 | 12 |
| | | | | 0 | 1 | 0 | 1 | 5 | | | | 75 | 59 | 43 | 27 | 11 |
| | | | | 0 | 1 | 1 | 0 | 6 | | | | 74 | 58 | 42 | 26 | 10 |
| | | | | 0 | 1 | 1 | 1 | 7 | | | | 73 | 57 | 41 | 25 | 9 |
| | | | | 1 | 0 | 0 | 0 | 8 | | | | 72 | 56 | 40 | 24 | 8 |
| | | | | 1 | 0 | 0 | 1 | 9 | | | | 71 | 55 | 39 | 23 | 7 |
| | | | | 1 | 0 | 1 | 0 | 10 | | | | 70 | 54 | 38 | 22 | 6 |
| | | | | 1 | 0 | 1 | 1 | 11 | | | | 69 | 53 | 37 | 21 | 5 |
| | | | | 1 | 1 | 0 | 0 | 12 | | | | 68 | 52 | 36 | 20 | 4 |
| | | | | 1 | 1 | 0 | 1 | 13 | | | | 67 | 51 | 35 | 19 | 3 |
| | | | | 1 | 1 | 1 | 0 | 14 | | | | 66 | 50 | 34 | 18 | 2 |
| | | | | 1 | 1 | 1 | 1 | 15 | | | | 65 | 49 | 33 | 17 | 1 |

Coding of line and column address for absolute positioning command:
IS4, ZLA, SPA, SAD (SAD = X'30')

Line addresses

## 3.3 Message format including field separators

Field separators define the attributes of the text they precede and are always located at the start of a field.

There are two types of field separator:

- field handling characters (FBZ)
- display control characters (ASZ).

A field handling character determines the field attributes of the texts it precedes and is valid until the next field handling character or until end of screen, while, independently of FBZ's, display control characters specify the video attributes of text segments. (In this context, a text segment may be a field or defined sections of a field, or may extend beyond a field boundary.) Field separators can be written to terminal memory via downline-loading from the host only; entry via the keyboard is not possible.

Field separators are identified by IS2 or IS3 code extension characters. (These characters are not stored in terminal memory.) The following assignment has been established:

IS2/... code extension character for identifying a field handling character (X'1E');
IS3/... code extension character for identifying a display control character (X'1D').

Up to 48 displayed field separators may be inserted between displayed characters (with no restriction on the number of displayable characters per line).

If additional field separators are inserted into a line already containing 48 separators, the terminal reacts as follows:

field separators inserted after separator 48 are ignored;
inserting a field separator before separator 48 causes the last separator in the line to be eliminated.

The assignment of field handling characters and display control characters is specified by setting switch 5 in terminal memory. There are two possibilities:

- both field handling characters and display control characters can be interpreted for video attribute differentiation;
- display control characters only are interpreted for video attribute differentiation.

The following field and video attributes can be produced by means of field separators:

Field attributes

| Field definition | Desig. | Remarks |
|---|---|---|
| Protected fields, sendable | PS | Field contents cannot be modified via the keyboard. They are included in transmission to host unless SAW = X'4F' and a MOD bit is set. |

Field attributes (cont.)

| Field definition | Desig. | Remarks |
|---|---|---|
| Protected fields, non-sendable | P | Field contents cannot be modified via the keyboard. They are not included in transmission to host if the fields are defined as protected and non-sendable unless SAW = X'4F' and a MOD bit is set. |
| Protected fields; addresses only are sent | PA | Provided SAW = X'41', X'45', X'4D' |
| Numeric fields | N | Only numerics, null characters, logical end-of-line, and the special symbols * + , . / may be entered in numeric fields via the keyboard. |
| Modified fields | MO | Created by:<br>• host action;<br>• keyboard entry<br>- valid character<br>- insert/delete characters<br>- LZF function (dependent on status of FST2 of PAR 00L);<br>• marking of markable fields (MOD bit + flash bit in display control character |
| MOD bit reset | | Dependent on status of FST2 in PAR 00L:<br>- after each transmission to and from host;<br>- by output of ESC FAZ in message;<br>- via LZF and LVD functions;<br>- by repeated marking. |
| Markable fields | MA | Markable fields can be marked via the MAR key. Fields can be visibly marked, i.e. flash, only in conjunction with a display control character, e.g. X'40'. This means that 2 field separators (1 field handling char.+ 1 display control character) must be located at the start of a markable field. |
| Non-markable fields | NM | These fields can be marked neither via the keyboard nor via a light pen. |
| Printable fields | PF | Printable fields can be output to printer. |
| Non-printable fields | NP | Output to printer not possible. |
| Variable fields | V | These fields can be freely written and modified. |

Video attributes

- **Flashing** — Characters flash at a frequency of approx. 2 Hz.
- **Underscore** — Characters are underscored.
- **Blanking** — Characters are blanked out. (Application example: non-displayed password)
- **Reduced intensity** — Characters are displayed at reduced intensity; contrast is adjustable.

### 3.3.1 Video attribute selection independent of field attributes

Field handling characters and display control characters are used separately and independently to determine the attributes of the text they precede. Field handling characters and display control characters combined may total no more than 48 in one line.

Field handling characters (IS2 sequences)

| Bit | Value | Field attributes |
|---|---|---|
| 1 | 1 | Protected field, non-sendable to host |
| 2 | 1 | Numeric field (but not in field mode) |
| 3 | 1 | Modified field ** |
| 4 | 1 | Markable field (but not in field mode) |
| 5 | 1 | Printable field |
| 6 | 1 | Protected field, sendable to host |
| 7 | 1 | For avoiding transmission control characters |

** Terminal-initiated; host initiation also possible.

Display control characters (IS3 sequences)

| Bit | Value | Video attributes |
|---|---|---|
| 1 | 1 | Flashing |
| 2 | 1 | Underscore |
| 3 | 1 | Blanking (video suppression) |
| 4 | 1 | Reduced intensity |
| 5 | 0 | Reserved |
| 6 | 0 | |
| 7 | 1 | For avoiding transmission control characters. |

Example

Text formatting with field handling characters (FBZ) and display control characters (ASZ) combined (48 max.). If video attributes are assigned to particular field definitions, up to 24 fields can be defined per line.

Data output from host:
IS4/ZLA3/SPA1/SAD0/ITB/IS3/ /FBZ/IS2/H/IS3/B/UND/IS2/A/IS3/H/Z/_/IS3/D/GEMISCHT

| Screen column | | | 1 | 2 | 3 | | | 4 | 5 | 6 | | | 7 | 8 | | | 9 | 10 | | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TEXT | | @ | F | B | Z | H | B | U | N | D | A | H | A | S | B | A | Z | ⌊ | | D | G | E | M | I | S | C | H | T |
| ID char. | IS3 | | | | | IS2 | IS3 | | | | IS2 | IS3 | | | IS2 | IS3 | | | | IS2 | | | | | | | | |

ASZ: Stand. Font, Normal Intensity

FBZ: Markable Field

ASZ: Underscore

FBZ: Protected Field

ASZ: Reduced Intensity

FBZ: Numeric Field

ASZ: Flashing

ASZ: Blanking

### 3.3.2 Video attribute selection in conjunction with field attributes

Switch S5, bit 4 must be set to 1.

In many applications, particular field attributes are always used in conjunction with particular video attributes. This assignment can be hardware-implemented in the 9750 by means of straps. This mode permits video attribute control to be handled by the field handling character: no display control characters are required. Thus, a line may contain the maximum number of field separators - 48 - in the form of field handling characters: up to 48 fields may be defined per line. (A display control character is interpreted in this mode if it is located immediately before the character to be defined or the start of field.)

| Bit | Value | Field attribute | | Video attribute |
|---|---|---|---|---|
| 1 | 1 | Protected field, non-sendable to host | A1 E1 | Normal intensity |
| 2 | 1 | Numeric field | A2 E2 | Flashing |
| 3 | 1 | Modified field | A3 | |
| 4 | 1 | Markable field | A4 E3 | Underscore/reverse vid. |
| 5 | 1 | Printable field | A5 E4 | Blanking |
| 6 | 1 | Protected field, sendable to host | A6 E5 | Reduced intensity |
| | | Protected field: bit 1 = 1 (see bit 1); bit 6 = 1 (see bit 6) | A7 | |
| 7 | 1 | For avoiding transmission control chars. | | |

------- Ex-works setting (alterable by DP maintenance personnel)

Of the seven field attributes (A1 - A7), up to five may be combined with the video attributes E1 - E5 on a one-to-one basis. If video attributes are combined with each other, e.g. E2 (reduced intensity) with E5 (flashing), the two video attributes react as a new attribute. This means that only four video attributes can be assigned to field attributes.

Example

Text formatting with field handling characters only.

Data output from host:
IS4/ZLA3/SPA4/SADO/ITB/FBZ/IS2/H/UND/IS2/A/AS/IS2/B/Z_VE/IS2/ /RKNÜPFT

| Screen column | 1 | 2 | 3 | | 4 | 5 | 6 | | 7 | 8 | | 9 | 10 | 11 | 12 | | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | F | B | Z | | H | U | N | D | | A | A | S | | B | Z | [ | V | E | | @ | R | K | N | Ü | P | F | T |
| ID char. | | | | IS1 | | | | IS2 | | | | | IS2 | | | | | | IS1 | | | | | | | | |

FBZ: Markable, Flashing

FBZ: Protected, Reduced Intensity

FBZ: Numeric

FBZ: Unprotected, Normal Intensity

## 3.4 Modifying message sections via host-to-terminal output

Messages on screen can be modified at any time by means of absolute positioning operations. Not only visible characters, but also invisible field separators can be modified, eliminated or overwritten.

### 3.4.1 Modifying a visible character

The following text in line 3 of the display is to be corrected.

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | ␣ | ␣ | D | D | T | ␣ | 9 | 7 | 5 | 0 | ␣ | ␣ | E | X | A | MPLE |
| Cursor pos.: random | | | | | | | | | | | | | | | | |

Data output from host: IS4/ZLA3/SPA10/SAD0/ITB/0

ID character
Line address (line 3)
Column address (column 10)
Distance address (0)
Control character (ISI)
Text character > 1 =

Corrected display:

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | . | . . . . . . |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | ␣ | ␣ | D | D | T | ␣ | 9 | 7 | 5 | 0 | ␣ | ␣ | E | X | A | M P L E |
| Cursor pos. | | | | | | | | | | ▷ | | | | | | |

## 3.4.2 Modifying field separators

Field separators can be modified by

- insertion,
- deletion,
- overwriting.

Modification can be handled only in a output sequence from the host as follows:

- output contains field separators only;
- output contains text and field separators;
- output contains text only (all field separators deleted).

### Inserting field separators (FTZ)

A field separator is inserted if there is no field separator of the same type to the left of the cursor position.

The following text is displayed in line 3.

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 1 | ␣ | E | X | A | M | P | L | E | S | ␣ | F | O | R | ␣ | E | N | T | E | R | I | N | G |
| Cursor pos. | | | | | | | ▷ | | | | | | | | | | | | | | | | | |
| FTZ pos. in memory | No field separators (FTZ) | | | | | | | | | | | | | | | | | | | | | | | |

### Example

Insertion of a single field handling character (FBZ) (IS2 sequence = FBZ), line 3, column 7

Data output from host: IS4/ZLA3/SPA7/SAD0/ITB/IS2

New text with field handling character (FBZ):

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 2 | ␣ | E | X | A | @ | M | P | L | E | S | ␣ | F | O | R | ␣ | E | N | T | E | R | I | N | G |
| ID char. | | | | | | | ▏ | | | | | | | | | | | | | | | | | | |
| Cursor pos. | | | | | | | | ▷ *) | | | | | | | | | | | | | | | | | |

*) With tracking cursor

Example

Insertion of a second field separator (FTZ) (IS3 sequence = display control character (ASZ)), line 3, column 7

Data output from host: IS4/ZLA3/SPA7/ITB/IS3/A

New text with field handling character and display control character:

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 3 | ␣ | E | X | A | @ | | A | M | P | L | E | S | ␣ | F | O | R | ␣ | E | N | T | E | R | I | N | G |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | | | | | | | | | | | | |
| Cursor pos. | | | | | | | | | | ▷ | | | | | | | | | | | | | | | | | |

Example

Insertion of two field separators (IS2 sequence = FBZ, IS3 sequence = ASZ), line 3, column 13

Data output from host: IS4/LA3/SPA13/SAD0/ITB/IS2/ /IS3/A

New text with field handling characters and display control characters:

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | 7 | 8 | 9 | 10 | 11 | 12 | | | | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 4 | ␣ | E | X | A | @ | | A | M | P | L | E | S | ␣ | @ | | A | F | O | R | _ | E | N | T | E | R | I | N | G |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | IS2 | | IS3 | | | | | | | | | | | | |
| Cursor pos. | | | | | | | | | | | | | | | | | | | ▷ | | | | | | | | | | | |

## Deleting field separators (FTZ)

### Deleting a single field separator

Field separators can be deleted individually by outputting IS2/NUL or IS3/NUL (NUL = X'00'). If there is no field separator of the specified type to the left of the specified position, the command has no effect. If there is no field separator at all, error message X'53' is set in character FEM of PAR 01E.

The following text is displayed in line 3.

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | 7 | 8 | 9 | 10 | 11 | 12 | | | | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 5 | ␣ | F | T | Z | | @ | | A | D | E | L | E | T | I | | B | | D | O | N |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | | IS2 | | IS3 | | | |
| Cursor pos. | | | | | | | | | | | | | | | | | | | | | ▷ | |

Example
Deletion of a single field separator (FTZ) (IS3 sequence = ASZ), line 3, column 13

Data output from host: IS4/ZLA3/SPA13/SAD0/ITB/IS3/NUL

New text:

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | 7 | 8 | 9 | 10 | 11 | 12 | | | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 6 | ␣ | F | T | Z | | @ | | A | D | E | L | E | T | I | | B | O | N |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | | IS2 | | | |
| Cursor pos. | | | | | | | | | | | | | | | | | | ▷ | | |

OR

Data output from host: IS4/ZLA3/SPA7/SAD0/ITB/Z/IS2/ /A

New text:

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | 7 | 8 | 9 | 10 | 11 | 12 | | | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 7 | ␣ | F | T | Z | | @ | D | E | L | E | T | I | | B | O | N |
| ID char. | | | | | | | IS2 | | | | | | | | IS2 | | | |
| Cursor pos. | | | | | | | | | | ▷ | | | | | | | | |

If there is no displayable character after the new field separator (IS2,@), the second field separator (IS3, A) is retained. If there is a displayable character, the second separator is deleted.

Deleting all field separators (FTZ) between two fields

The field separators are deleted if the last character of the previous field and the first character of the next field are to be overwritten.

The following text is displayed in line 3.

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | | 7 | 8 | 9 | 10 | 11 | 12 | | | | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 8 | ␣ | E | X | A | | B | | D | M | P | L | E | S | ␣ | @ | | A | F | O | R | ␣ | D | E | L | E | T | I | O | N |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | | IS2 | IS3 | | | | | | | | | | | | | | |
| Cursor pos. | | | | | | ▷ | | | | | | | | | | | | | | | | | | | | | | | | | |

Example

Deletion of all field separators (IS2 sequence = field handling character (FBZ), IS3 sequence = display control character (ASZ)), line 3, column 7

Data output from host: IS4/ZLA3/SPA6/ITB/I/S

New text without specified field handling character and display control character:

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | | | | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 0 | 9 | ␣ | E | X | A | M | P | L | E | S | ␣ | @ | | A | F | O | R | ␣ | D | E | L | E | T | I | O | N |
| ID char. | | | | | | | | | | | | | IS2 | IS3 | | | | | | | | | | | | | |
| Cursor pos. | | | | | | | | ▷ | | | | | | | | | | | | | | | | | | | |

Overwriting field separators (FTZ)

Overwriting individual field separators

Field separators are overwritten when, during host-to-terminal transmission, the last character at the end of a field is overwritten and then a new field separator is output.

The following text is displayed in ine 3.

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | | 7 | 8 | 9 | 10 | 11 | 12 | | | | | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 1 | 0 | ⌴ | 0 | V | E | | @ | | A | R | W | R | I | T | E | | @ | | A | F | T | Z | S |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | | IS2 | | IS3 | | | | | |
| Cursor pos. random | | | | | | | | | | | | | | | | | | | | | | | | |

Example

Overwriting of a field separator of the same type (IS2 or IS3), line 3, column 13

Data output from host: IS4/ZLA3/SPA13/SAD0/ITB/IS2/H

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | 7 | 8 | 9 | 10 | 11 | 12 | | | | 13 | 14 | 15 | 16 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 1 | 1 | ⌴ | 0 | V | E | | @ | A | R | W | R | I | T | E | | H | | A | F | T | Z | S |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | IS2 | | IS3 | | | | | |
| Cursor pos. | | | | | | | | | | | | | | | | | | | | ▷ | | | |

If there is no displayable character after the new field separator (IS2, H), the second field separator (IS3, A) is retained. If there is a displayable character, the second separator is deleted. If two field separators are output, the new field separator overwrites the second separator (IS3, A) too.

## Overwriting all field separators (FTZ) between two fields

A field separator can be overwritten if there is a separator of the same type to the left of the cursor position. The position of the correct field separator is ascertained by the terminal's control program.

The following text is displayed in line 3.

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | 7 | 8 | 9 | 10 | 11 | 12 | | | | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 1 | 4 | ␣ | E | X | A | | H | | A | M | P | L | E | S | ␣ | | @ | | A | F | O | R | ␣ | O | V | E | R | W | R | I | T |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | | IS2 | | IS3 | | | | | | | | | | | | | |
| Cursor pos. | | | | | | | | | | ▷ | | | | | | | | | | | | | | | | | | | | | | |

Example

Overwriting of two field separators (IS2 sequence = field handling character (FBZ), IS3 sequence = display control character (ASZ)), line 3, column 7

Data output from host: IS4/ZLA3/SPA7/ITB/IS2/B/IS3/D

New text with FBZ/ASZ:

| Screen column | 1 | 2 | 3 | 4 | 5 | 6 | | | | 7 | 8 | 9 | 10 | 11 | 12 | | | | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Text | 1 | 5 | ␣ | E | X | A | | B | | D | M | P | L | E | S | ␣ | | @ | | A | F | O | R | ␣ | O | V | E | R | W | R | I | T |
| ID char. | | | | | | | IS2 | | IS3 | | | | | | | | IS2 | | IS3 | | | | | | | | | | | | | |
| Cursor pos. | | | | | | | | | | ▷ | | | | | | | | | | | | | | | | | | | | | | |

It is, of course, possible to modify only one of the two field separators.

## 3.5 Message format including device functions

Device functions can be initiated both via the keyboard and downline via the host. (The restrictions noted in the relevant tables must be observed.) A device function is implemented in different ways, depending on the functional status and mode status of the terminal at the time of initiation. After power-up and after the reset function has been initiated, the device is in its defined initial state.

### Entering the device functions

Device functions can be initiated by means of two-character sequences in a message header or two/three-character sequences in parameter entries or message sections.

| | |
|---|---|
| Coding of two-character sequences: | ESC sequences from code table, column 2 (entry in PAR 00L or PAR 00D) |
| Coding of three-character sequences: | ESC/SP sequences from code table, columns 3 - 6 (entry in PAR 00D only) |

No identifying ESC character is required when message headers or parameter entries are used. If a message header is used, only the two-character sequences are acted upon. In the case of parameter entries, two and three-character sequences are accommodated in character GEF1 (PAR 00D) and GEF2 (PAR 00D) respectively. (The character SP is used here as an identifier for a three-character sequence in character GEF1 of PAR 00D.) The functions LVA and WDH are exceptions to this in that they can only be accommodated within the text section, and for this positioning commands are required.

### Implementing the device functions

In the case of parameter entries, the corresponding is not implemented until text transmission has been completed. If entered as an ESC sequence within a message block, the function is implemented while the text is displayed on screen.

DÜ, K, F and LA functions, as well as P functions containing one of the send functions mentioned, may only be specified - in parameter entries - as device functions. (If entered as ESC sequences within a message block, these functions will be ignored and an error message will be set in character FEM of PAR 01E.

In the case of transmission from terminal to host, the code of the send key actuated (K1 - K14, F1 - F5, DÜ) is entered in character CDS (char. 7) of PAR 00E.

The following device functions can be implemented on the 9750 Data Display Terminal:

- cursor positioning,
- text shift and editing,
- erase,
- send,
- intra-system data interchange,
- programmable key functions,
- special functions.

The data display terminal's device functions can be activated both from the keyboard and, with limitations, remotely from the host.

## 3.5.1 Device function coding

**Assignment of device and function control characters in accordance with status of ESC/GEF**

| Bits b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Row \ Column | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| | | | | | | | | | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| Parity bit | | | | 0 | 0 | 0 | 0 | 0 | | | | AKA | SMR | PAR | RU | BRS |
| | | | | 0 | 0 | 0 | 1 | 1 | | | | * | SML | LA1 | P2 | LZE |
| | | | | 0 | 0 | 1 | 0 | 2 | | | | * | SMO | LA2 | SBA | |
| | | | | 0 | 0 | 1 | 1 | 3 | | | | * | SMU | K1 | P1 | |
| | | | | 0 | 1 | 0 | 0 | 4 | | | | * | SNZ | K2 | LSP | |
| | | | | 0 | 1 | 0 | 1 | 5 | | | | * | SZA | K3 | LVD | |
| | | | | 0 | 1 | 1 | 0 | 6 | | | | * | TAR | K4 | DÜ | |
| | | | | 0 | 1 | 1 | 1 | 7 | | | | * | TAL | K5 | | |
| | | | | 1 | 0 | 0 | 0 | 8 | | | | * | EFG | | P3 | |
| | | | | 1 | 0 | 0 | 1 | 9 | | | | * | AFG | | P4 | |
| | | | | 1 | 0 | 1 | 0 | 10 | | | P5 | K14 | AFZ | | P | |
| | | | | 1 | 0 | 1 | 1 | 11 | | ESC | P6 | K13 | EFZ | F1 | SDZ | |
| | | | | 1 | 1 | 0 | 0 | 12 | | | | K12 | LZF | F2 | MAR | |
| | | | | 1 | 1 | 0 | 1 | 13 | | | | K11 | K6 | F3 | FAZ | |
| | | | | 1 | 1 | 1 | 0 | 14 | | | | K10 | K7 | F4 | RS | |
| | | | | 1 | 1 | 1 | 1 | 15 | | | | K9 | K8 | F5 | VA | |

* Reserved for 8121 and 8122 Printers.

Assignment of device and function control characters in accordance with status of ESC/GEF (cont.)

| Bits b8 | b7 | b6 | b5 | b4 | b3 | b2 | b1 | Row \ Column | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| | | | | | | | | | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| | | | | | | | | | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Parity bit | | | | 0 | 0 | 0 | 0 | 0 | | | SP | | P7 | AM | FON | |
| | | | | 0 | 0 | 0 | 1 | 1 | | | | | P8 | DÜ2 | PAK | |
| | | | | 0 | 0 | 1 | 0 | 2 | | | | | P9 | SS | WDH | |
| | | | | 0 | 0 | 1 | 1 | 3 | | | | | P10 | | LVA | |
| | | | | 0 | 1 | 0 | 0 | 4 | | | | | P11 | | | |
| | | | | 0 | 1 | 0 | 1 | 5 | | | | | P12 | | | |
| | | | | 0 | 1 | 1 | 0 | 6 | | | | | P13 | | | |
| | | | | 0 | 1 | 1 | 1 | 7 | | | | | P14 | | | |
| | | | | 1 | 0 | 0 | 0 | 8 | | | | | P15 | | | |
| | | | | 1 | 0 | 0 | 1 | 9 | | | | | P16 | | | |
| | | | | 1 | 0 | 1 | 0 | 10 | | | | | P17 | | | |
| | | | | 1 | 0 | 1 | 1 | 11 | | ESC | | | P18 | LA3 | | |
| | | | | 1 | 1 | 0 | 0 | 12 | | | | | P19 | LA4 | | |
| | | | | 1 | 1 | 0 | 1 | 13 | | | | | P20 | LA5 | | |
| | | | | 1 | 1 | 1 | 0 | 14 | | | | | | LA6 | | |
| | | | | 1 | 1 | 1 | 1 | 15 | | | | | | LA7 | | |

## 3.5.2 Cursor function description

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in head. or par. entr.* | Programmable in text |
|---|---|---|---|---|---|---|
| SMR  | X'40' | Cursor Right | Cursor jumps one char. forwards. At end of screen, cursor jumps to start of screen. | If autotab (ATAB) is activated and bit 4 of FST1 in PAR 00L is 0, cursor skips over protected, non-markable fields, automatically moving to first col. of next or preceding unprotected field.<br><br>If autotab (ATAB) is activated and bit 4 of FST1 in PAR 00L is 1, cursor is positioned in accordance with function initiated, irrespective of field definition. | X'40' | ESC;'40' |
| SML  | X'41' | Cursor Left | Cursor jumps one char. backwards. At start of screen, cursor jumps to end of screen. | | X'41' | ESC;'41' |
| SMO  | X'42' | Cursor Up | Cursor jumps one line upwards, remaining in same col. Upon reaching line 1, cursor is positioned in line 24. | | X'42' | ESC;'42' |
| SMU  | X'43' | Cursor Down | Cursor jumps one line downwards, remaining in same col. Upon reaching line 24, cursor is positioned in line 1. | | X'43' | ESC;'43' |
| SNZ  | X'44' | Cursor To Start Of Next Line | Cursor jumps to start of next line below. Upon reaching line 24, cursor jumps to start of screen. | | X'44' | ESC;'44' |
| SZA  | X'45' | Cursor To Start Of Line | Cursor jumps to start of line. | | X'45' | ESC;'45' |
| SBA  | X'62' | Cursor To Start Of Screen | Cursor jumps to start of screen. | | X'62' | ESC;'62' |

* Three-character sequences can be accommodated in parameter entries only.

Cursor function description (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in head. or par. entr. * | Programmable in text |
|---|---|---|---|---|---|---|
| SDZ  | X'6B' | Cursor To Start Of Line Above | Cursor jumps to col. 1 of line above. Upon reaching line 1, cursor jumps to start of line 24. | | X'6B' | ESC;'6B' |
| TAR  | X'46' | Tab Right | Cursor jumps forward 16 positions at a time (col. 1, 17, 33, 49, 65). Upon reaching col. 65 of line 24, cursor jumps to start of screen. | Cursor jumps forward to col. 1 of next modifiable field. | X'46 | ESC;'46' |
| TAL  | X'47 | Tab Left | Cursor jumps backwards 16 positions at a time (col.65, 49, 33, 17, 1). Upon reaching start of screen, cursor jumps to col. 65 of line 24. | Cursor jumps backwards to col. 1 of next modifiable field. | X'47' | ESC;'47' |

* Three-character sequences can be accommodated in parameter entries only.

### 3.5.3 Text shift and editing functions

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| EFG | X'48' | Insert Characters | Characters are moved forwards, from cursor position, by the number of chars. inserted.<br><br>Null chars. in the text are overwritten. If there are no more null chars. left, all subsequent chars. are moved forwards.<br><br>The function remains effective beyond the end of a line. Characters shifted beyond end of screen are lost.<br><br>If, during insertion, a LZE (logical end of line) char. reaches the end of a physical line, the characters in this line and in the following lines are moved forwards until the conditions for logical end of line are met.<br><br>Insert Characters mode is deactivated via the RS key. | As for non-formatted mode, but restricted to an unprotected field.<br><br>Characters shifted beyond the end of a field are lost.<br><br>Display control chars. are not affected.<br><br>Inserting a char. via the keyboard causes the MOD bit to be set in the field handling character. | X'48' | ESC;'48' |
| AFG | X'49' | Delete Character | The character at the cursor position is deleted; subsequent chars. are shifted backwards one column.<br><br>Positions becoming free are occupied by null chars.<br><br>The function is effective up to end of screen. | As for non-formatted mode, except that the function is not effective beyond the end of an unprotected field.<br><br>Deletion via the keyboard causes the MOD bit to be set in the field handling char. | X'49' | ESC;'49' |

Text shift and editing functions (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| EFZ | X'4B' | Insert Line (can be blocked for user via char. FST1 of PAR 00L) | If bit 2 of FST1 in PAR 00L is 0, all lines below and including the line the cursor is in are shifted downwards.<br><br>If bit 2 of FST1 in PAR 00L is 1, the above applies. In addition, the last line is copied to the cursor line. | As for non-formatted mode, but note the following.<br><br>Depending on status of bit 2 of FST1 in PAR 00L, format data is handled as follows:<br><br>bit 2 = 0<br>Field separators and protected data from the last line are copied to the cursor line and unprotected data are replaced by null chars.<br><br>bit 2 = 1<br>Field separators and protected data from the last line are copied to the cursor line. | X'4B' | ESC;'48' |
| AFZ | X'4A' | Delete Line (can be blocked for user via char. FST1 of PAR 00L) | If bit 3 of FST1 in PAR 00L is 0, all lines below and and including the line the cursor is in are shifted upwards.<br><br>If bit 3 of FST1 in PAR 00L is 1, the above applies. In addition, the cursor line is copied to the last line (scroll-up). | As for non-formatted mode, but note the following.<br><br>Depending on status of bit 2 of FST1 in PAR 00L, format data are handled as follows:<br><br>bit 2 = 0<br>Field separators and protected data are copied from the cursor line to the last line.<br><br>bit 2 = 1<br>Field separators and all data in the cursor line are copied to the last line. | X'4A' | ESC;'4A' |

Text shift and editing functions (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| RU | X'60' | Roll-Up Function (can be blocked for user via char. FST1 in PAR 00L) | Lines 2 - 24 are each moved one line upwards. The characters in line 1 are lost. Null chars. are inserted in line 24. Cursor is positioned at start of line 24. | As for non-formatted mode. Field separators are also moved.<br><br>Line 24 contains no field separators.<br><br>The field separators in line 1 are lost. | X'60' | ESC;'60' |
| RUB | --- | Roll-Up Mode (activated and deactivated via char. FST1 in PAR 00L) | After the last write function for line 24 has been performed, lines 2 - 24 are each moved one line upwards.<br><br>The characters in line 1 are lost. Null characters are inserted in line 24. Cursor is at start of line 24.<br><br>If a logical end of line (LZE) is entered in line 24, the rest of line 24 is overwritten with null chars., the roll-up function is initiated, and the cursor is positioned at the start of the new line 24.<br><br>Once 23 lines have been moved upwards, the RU function is disabled. It is not enabled until the RS key has been depressed.<br><br>The line counter is reset as a result of a dialog, too. | As for non-formatted mode (inclusive of all field separators)<br><br>NB<br>If RU mode and autotab (ATAB) are used, the last character on screen must be in an unprotected field. | --- | --- |

### 3.5.4 Erase functions

| Function desig. | ASCII rep. | Function | Implementation, non-formattted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| LSP | X'64' | Clear Memory (host-initiated in PAR 00D or PAR 00L (GEF1) OR keyboard-initiated via LSP key, which can be blocked via FST2 in PAR 00L) | Erase<br>• from start to end of screen (overwrite with null chars.)<br>• PAR 00L<br>• PAR 10L - PAR 70L (depending on status of FST2 in PAR 00L) | As for non-formatted mode (inclusive of all field separators) | X'64' | |
| | | LSP (init. via ESC seq. in text) | Erase from cursor to end of screen (no repositioning of cursor). Parameter ranges are not erased. | As for non-formatted mode (inclusive of all field separators). | --- | ESC;'64' |
| LVD | X'65' | Erase Variable Data | Erase from cursor up to and including an end marker or to end of screen (overwrite with null chars.) After completion of the erase function, the cursor is in its home position.<br><br>The erase limits are specified by bit 6 of character FST2 in PAR 00L:<br><br>bit 6 = 0: Erase from start of screen to end marker.<br>bit 6 = 1: Erase from start to end of screen. | Erase unprotected fields from cursor up to and including an end marker or to end of screen (overwrite with null chars.) After completion of the erase function, the cursor is in its home position. | | ESC;'65' |

Erase functions (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| LZF | X'4C' | Erase To End Of Line Or End Of Field | Erase from cursor to end of line or to an end marker in this line. (The end marker is also erased.) After the erase operation, the cursor is in its home position. | All characters in an unprotected field are erased from cursor to end of field or up to and including an end marker in the same field. The cursor position is unchanged.<br><br>The status of the modify and flash bits (MOD and BLINKEN) is specified in bit 5 of char. FST2 in PAR 00L. | X'4C' | ESC;'4C' |
| LVA | X'63' | Erase Variable Data To Address | All variable data between the cursor position and the address specified are erased.<br><br>The position address in the second sequence specifies the position of the first character that is not to be erased. If the position address is missing, the next two characters following the LVA function will be used as the position address. If no address is specified or if the address specified is invalid, the function will not be performed.<br><br>Modify (MOD) bits remain unchanged. | | | ESC;<br>'20';'63';<br>ZLA;SPA |

## 3.5.5 Send functions

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| DÜ1 | X'66' | Send | When a send function is initiated, a send request is sent to the host. SAN is displayed in line 25.<br><br>SAN can be reset via the RS key. Message transmission is in accordance with the entries in char.1 (SAW1) in parameter range PAR 00L.<br><br>The code of the send key used is entered in char. 7 (CDS) in PAR 00E. | | X'66' | |
| DÜ2 | X'51' | Send | Implementation as for DÜ1. Message transmission is in accordance with the entries in char. 6 (SAW2) in parameter range 01L.<br><br>The code of the send key used is entered in char. 7 (CDS) in PAR 00E. | | X'20'; X'51' (three-char. seq.) | |
| K1 | X'53' | Send Short Message (Telegram) | Initiated via keys K1 - K3 | When a K function is initiated, the corresponding code is entered in char. 7 (CDS) in PAR 00E. | X'53' | |
| K2 | X'54' | | | | X'54' | |
| K3 | X'55' | | | | X'55' | |
| K4 | X'56' | | Initiated via ESC key and one of the keys on the right: V W M N O ? > = < ; : | A short message (telegram) may comprise PAR 00E, PAR 01E and a text ID.<br><br>A K function can also be initiated if the keyboard is blocked. If a K function is to be initiated from the host, the appropriate code must be entered in char. 1 (GEF1) in PAR 00D. | X'56' | |
| K5 | X'57' | | | | X'57' | |
| K6 | X'4D' | | | | X'4D' | |
| K7 | X'4E' | | | | X'4E' | |
| K8 | X'4F' | | | | X'4F' | |
| K9 | X'3F' | | | | X'3F' | |
| K10 | X'3E' | | | | X'3E' | |
| K11 | X'3D' | | | | X'3D' | |
| K12 | X'3C' | | | | X'3C' | |
| K13 | X'3B' | | | | X'3B' | |
| K14 | X'3A' | | | | X'3A' | |

Send functions (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in | Programmable in |
|---|---|---|---|---|---|---|
| F1<br>F2<br>F3 | X'5B'<br>X'5C'<br>X'5D' | Send | Initiated via keys F1 - F3 | When an F function is initiated, the corresponding code is entered in character 7 (CDS) of PAR 00E. Together with the displayed message, it is sent to the host as specified in the send command (PAR 00L, char. 1 (SAW1)).<br><br>If an F function is to be initiated from the host, the appropriate code must be entered in character 1 (GEF1) in PAR 00D. | X'5B'<br>X'5C'<br>X'5D' | |
| F4<br>F5 | X'5E'<br>X'5F' | | Initiated via ^ ESC key and one of _ the keys on the right: | | X'5E'<br>X'5F' | |

### 3.5.6 Intra-system data interchange functions

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR |
|---|---|---|---|---|---|
| LA1 | X'51' | Local Data Inter-change | Functions LA1 - LA4 are initiated via the dedicated keys directly. | | X'51' |
| LA2 | X'52 | | | | X'52' |
| LA3 | X'5B' | | | | X'20';'5B' |
| LA4 | X'5C' | | Functions LA5 - LA7 are initiated via ESC/SP/LAx sequences. | | X'20';'5C' |
| LA5 | X'5D' | | | | X'20';'5D' |
| LA6 | X'5E' | | Functions LA1 - LA7 can also be initiated from the host. | | X'20';'5F' |
| LA7 | X'5F' | | | | X'20';'5F' |
| | | | Depending on the address specifications in parameter ranges PAR 10L/D through PAR 70L/D, an output device attached locally to the data display terminal or a printer terminal connected to the same cluster controller will be selected for data interchange. | | |
| | | | | All printable fields from the cursor position to an end marker or to end of screen are printed out.<br><br>Printer control characters in printable fields are interpreted in accordance with the specifications in the print command (PAR 10L/D - PAR 70L/D; see Section 5.3.4). | |

## 3.5.7 Programmable key functions

| Func-desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| P | X'6A' | Activate Programming Mode | The P function activates programming mode, in which the P registers can be programmed. If this mode is not activated, the P registers cannot be programmed. After initiation, TASTE Pn=? (Key Pn=?) is displayed in the 9750's status line. After selecting the Pn function required, the corresponding register range is displayed and released for programming, provided no programming inhibit flag is set in character 5 (FST7) in parameter range PAR 01L. (Programming inhibits are only valid for keys P1 - P5.) After the last location in the register range has been written to, programming mode is deactivated by reinitiating the P function.<br><br>The procedure for the downline-loading of the P registers from the host is the same as for keyboard (manual) loading. | | | ESC;'6A' |
| P1 | X'63' | Select P Register (P1 - P20) | If programming mode has been activated, selecting one of the P function keys (P1 - P20) or an appropriate ESC sequence causes the range required to be displayed. The cursor is positioned at the start of the register range.<br><br>Storage capacity and entry capability<br>Up to 16 characters can be entered in a P register. Storable characters: alphanumerics, device control characters and field separators, e.g. ESC, IS2, IS3, IS4 sequences. (The characters are not interpreted during programming mode.) If an entry has to be corrected, a null character can be keyed in, the cursor being backspaced one position.<br><br>Chaining of P functions<br>In order to chain P functions, the code of a different P function may be entered in the P register already selected. (The new P function must have a higher value than the P register to be programmed.) After a valid P function has been entered, programming is terminated and the new register range accepted. | | X'63' | ESC;'63' |
| P2 | X'61' | | | | X'61' | ESC;'61' |
| P3 | X'68' | | | | X'68' | ESC;'68' |
| P4 | X'69' | | | | X'69' | ESC;'69' |
| P5 | X'2A' | | | | X'2A' | ESC;'2A' |
| P6 | X'2B' | | | | X'2B' | ESC;'2B' |
| P7 | X'40' | | | | X'20';'40' | ESC;'20';'40' |
| P8 | X'41' | | | | X'20';'41' | ESC;'20';'41' |
| P9 | X'42' | | | | X'20';'42' | ESC;'20';'42' |
| P10 | X'43' | | | | X'20';'43' | ESC;'20';'43' |
| P11 | X'44' | | | | X'20';'44' | ESC;'20';'44' |
| P12 | X'45' | | | | X'20';'45' | ESC;'20';'45' |
| P13 | X'46' | | | | X'20';'46' | ESC;'20';'46' |
| P14 | X'47' | | | | X'20';'47' | ESC;'20';'47' |
| P15 | X'48' | | | | X'20';'48' | ESC;'20';'48' |
| P16 | X'49' | | | | X'20';'49' | ESC;'20';'49' |
| P17 | X'4A' | | | | X'20';'4A' | ESC;'20';'4A' |
| P18 | X'4B' | | | | X'20';'4B' | ESC;'20';'4B' |
| P19 | X'4C' | | | | X'20';'4C' | ESC;'20';'4C' |
| P20 | X'4D' | | | | X'20';'4D' | ESC;'20';'4D' |

Programmable key functions (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| | | | Keyboard initiation<br>When a Pn function (P1 - P20) is initiated, the contents of the corresponding register range are interpreted. Char. FST6 in PAR 01L may contain a specific user inhibit (individually for each of the keys P1 - P5) or a general user inhibit (for the keys P6 - P20 taken as a block).<br><br>NB<br>If the register range selected contains a send function, bit 5 of character 5 (FST3) in PAR 00L must be set at 1 (dialog mode; see Section 2.4.5).<br><br>Programming inhibit<br>If the P register is reserved for the host, actuating a Pn function key will cause DVA Pn (Host Pn) to be displayed, i.e. no characters can be entered. (Characters entered by the host will be processed.) A programming language may be specified in char. FST7 in PAR 01L: individually for each of the keys P1 - P5, and jointly for the keys P6 - P20 taken as a block.<br><br>Host initiation<br>Host initiation via device control characters in the message or in PAR 00D. If the P register contains a function that is invalid during host-to-terminal transmission, such as a DÜ, K, F or LA function, it will not be interpreted and implementation of the Pn function will continue.<br><br>Programming mode termination:<br>- by reinitiating the P function;<br>- when the last location in the register range has been written to. | | | |

### 3.5.8 Special functions

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| FAZ | X'6D' | Set Field Separators To Initial State | | The MOD bits (bit 3) in all field handling chars. and the BLINKEN bits (bit 1) in the display control chars. of marked fields are cleared. It can be specified in char. FST2 in PAR 00L whether this function is to be initiated upon reception of a device function command only or automatically after each output from the host. | X'6D' | ESC;'6D' |
| MAR | X'6C' | Mark Field | A MAR entry will be ignored because the function is effective in formatted mode only. | Markable fields start with field handling chars. (bit 4 = 1) and display control chars.<br><br>A field can be marked if it contains the cursor and the MSR function is initiated. (MOD bit in field separator is set.) The marked field starts flashing as a visual acknowledgement.<br><br>Repeated marking cancels the function for protected fields, or activates it. Unprotected fields remain marked.<br><br>Depending on the status of char. FST2 in PAR 00L, the BLINKEN (flash) bits in the fields marked by a display control char. are reset from the host. | X'6C' | ESC;'6C' |

Special functions (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| PAR | X'50' | Modify Parameter Range | Can be reset by initiating the RS function or an LA function.<br><br>NB: A message must not contain a PAR function. | | X'50' | ESC;'50' |
| RS | X'6E' | Reset | Resets error flags and function statuses without modifying memory contents.<br><br>Send requests set by acknowledgements cannot be reset by means of the RS function. | | X'6E' | ESC;'6E' |
| VA | X'6F' | Clear Down Connection | This function is effective for standalone data display terminals only. After completion of a DÜ, P or LA function, the connection is cleared by the terminal. | | X'6F' | ESC;'6F' |
| AM | X'50' | Start Marker | Starting from and including the start marker, unprotected fields (plus null chars.) up to cursor are sent, provided the SAW (Send) char. is appropriately coded.<br><br>The start marker is interpreted only in the case of transmission from terminal to host.<br><br>X'48' must be specified in PAR 00L (SAW1) or PAR 01L (SAW2).<br><br>The last start marker before the cursor is always interpreted. If a start marker is not specified in the SAW char., the send function starts at start of screen.<br><br>A start marker at end of screen has no effect. | | X'50' | ESC;'20';'50' |
| EM | X'19' | End Marker | The end marker permits the following functions to be delimited:<br>- message printout;<br>- erase operations;<br>- transm. from terminal to host if the appropriate send command (SAW) is entered in parameter range PAR 00L.<br><br>In the case of transmission from terminal to host, the end marker must be located in a sendable field. | | | X'19' |

Special functions (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| BEL | X'07' | Visual Alarm | Upon reception of the control character BEL, text BEL flashes in the status line. (If the appropriate facility is installed, an audible signal will also be emitted.)<br><br>BEL can be used to identify output from the host, for example.<br><br>BEL is reset by means of the BRS function. | | --- | X'07', as a separate text block without header, or before start or end sequence without STX or ETX. |
| BRS | X'70' | Reset | Resets the alarm function BEL. | | X'70' | ESC;'70' |
| AKA | X'30' | Audible Alarm | If installed, an audible alarm is emitted when the control character AKA is received. BEL is displayed, too.<br><br>AKA is issued at the end of a host-to-terminal transmission sequence in order to draw the user's attention to the fact that output has terminated.<br><br>BEL is reset by means of the BRS or RS function. | | X'30' | ESC;'30' |
| SS | X'52' | Set Cursor | It can be specified in char. 5 (FST3) in parameter range PAR 00L whether, during host-to-terminal transmission, the cursor is to track the message as it is received, or remain stationary. When the SS function is output, the cursor is set, or positioned at the current write position. | | X'20'; '52' | ESC;'20';'52' |

Special functions (cont.)

| Function desig. | ASCII rep. | Function | Implementation, non-formatted mode | Implementation, formatted mode | Programmable in PAR | Programmable in text |
|---|---|---|---|---|---|---|
| WDH | X'62' | Repeat | A character is repeated up to a specified line/column address. This function is equivalent to outputting the particular number of characters, e.g. field separators are overwritten.<br><br>The function is effective from the cursor position to the screen line/column position specified (ZLA, SPA).<br><br>If invalid addresses are specified, the function will not be implemented. If the specified address is before the current write position, the repeat function is implemented beyond the end of screen up to the address specified. If the specified address corresponds to the current write position, the entire display buffer is filled with prespecified characters.<br><br>Column/line addresses must be coded in accordance with the table in the Appendix. | | --- | ESC;'20';'62' ZLA;SPA; char. |
| LZE | X'71' | Logical End Of Line | Logical lines begin at start of line and end with LZE, and may occupy part of a physical line or several lines.<br><br>The cursor can reach all cursor positions.<br><br>All character after a LZE are erased up to end of line. | Logical lines begin at start of field and end with LZE, and may occupy part of a physical line or several lines.<br><br>The cursor can reach all cursor positions.<br><br>All characters except field separators are erased up to end of field or end of line. | | ESC;'20';'71' |

# 4 Message block formation

The formation of message blocks reduces response times. When a blocking character is recognized, internal processing of the message blocks already in the send buffer is performed parallel to data output.

A message can be divided into message blocks (by means of ETX or ETB characters) and message sections (by means of ITB characters). Messages larger than 7168 characters cannot be stored in their entirety in the send buffer. Repeated attempts to buffer an excessively large message or message block will result in a negative acknowledgement being generated.

In a configuration with the BS1000 Operating System, block formation must be carried out by the user irrespective of whether the 9750 Data Display Terminal is connected to the host via an

- 8170 Cluster Controller (local) with NEA1,
- 9683 or 9687 Front-End Processor with AMS1,
- 9684 or 9685 Front-End Processor with NEA2,
- 9674 or 9675 Remote Front-End Processor with NEA2, or a
- 9661, 9663 or 9665 Terminal Computer with NEA2

and irrespective of which communication control software is interposed, e.g. ASMUS. Automatic block formation is implemented in the BS2000 Operating System from Version 3.0 onwards.

# 5 Intra-system data interchange

Intra-system data interchange means that the screen contents of a 9750 Data Display Terminal can be output to a printer connected locally to the 9750 or to a printer terminal attached to the same cluster controller.

This mode can be initiated manually (via an LAx key) or remotely from the host (by means of an LAx function entry in the parameter ranges).

In the event of a printer or printer terminal error/fault during transmission from the host, selection sequences are rejected by the 9750 by means of WABT.

If "mandatory dialog" is activated in character FST8 (PARAM 0 or PAR00L), i.e. bit 5 = 1, keyboard entry is inhibited during output to a printer terminal. Keyboard entry can be enabled by means of an acknowledgement from the printer terminal. For this, an acknowledgement must be requested in character ADA (PAR 10L/D - PAR 70L/D).

## 5.1 Initiation of send functions

### 5.1.1 Initiation by the host

If intra-system data interchange is to be initiated by the host, the requisite device function (LA1 - LA7) must be entered in character GEF1 in parameter range PAR 00D. The 9750 interprets the appropriate parameter range (PAR 10D - PAR 70D) in accordance with the LA function specified.

In addition, character 4 (LAP1) in parameter range PAR 00D can be used to specify whether the message is to be output via the 9750's screen or - in non-user-transparent bypass mode - via its buffer storage.

### 5.1.2 Initiation by the user

Keys LA1 - LA4 are assigned to parameter ranges PAR 10L - PAR 40L. Actuating one of these LA keys causes the corresponding parameter range to be interpreted. In addition, the functions LA5 - LA7 (PAR 50L - PAR 70L) can be implemented as ESC sequences. (They can also be programmed via P keys.)

When the data display terminal is powered up, parameter range PAR 10L - LA1 key - is assigned to the local printer.

Bypass mode is not possible in the case of user initiation.

## 5.2 Use of acknowledgement characters

Acknowledgements inform the host whether a print operation has terminated positively (error-free) or negatively (errored). In the case of data interchange with a local printer, acknowledgements are generated by the data display terminal. In the case of internal traffic, acknowledgements are generated by the printer terminal, passed on to the data display terminal, and then sent to the host.

Acknowledgements are not generated by the data display terminal until the entire text has been printed. They are then sent to the host in the next polling sequence.

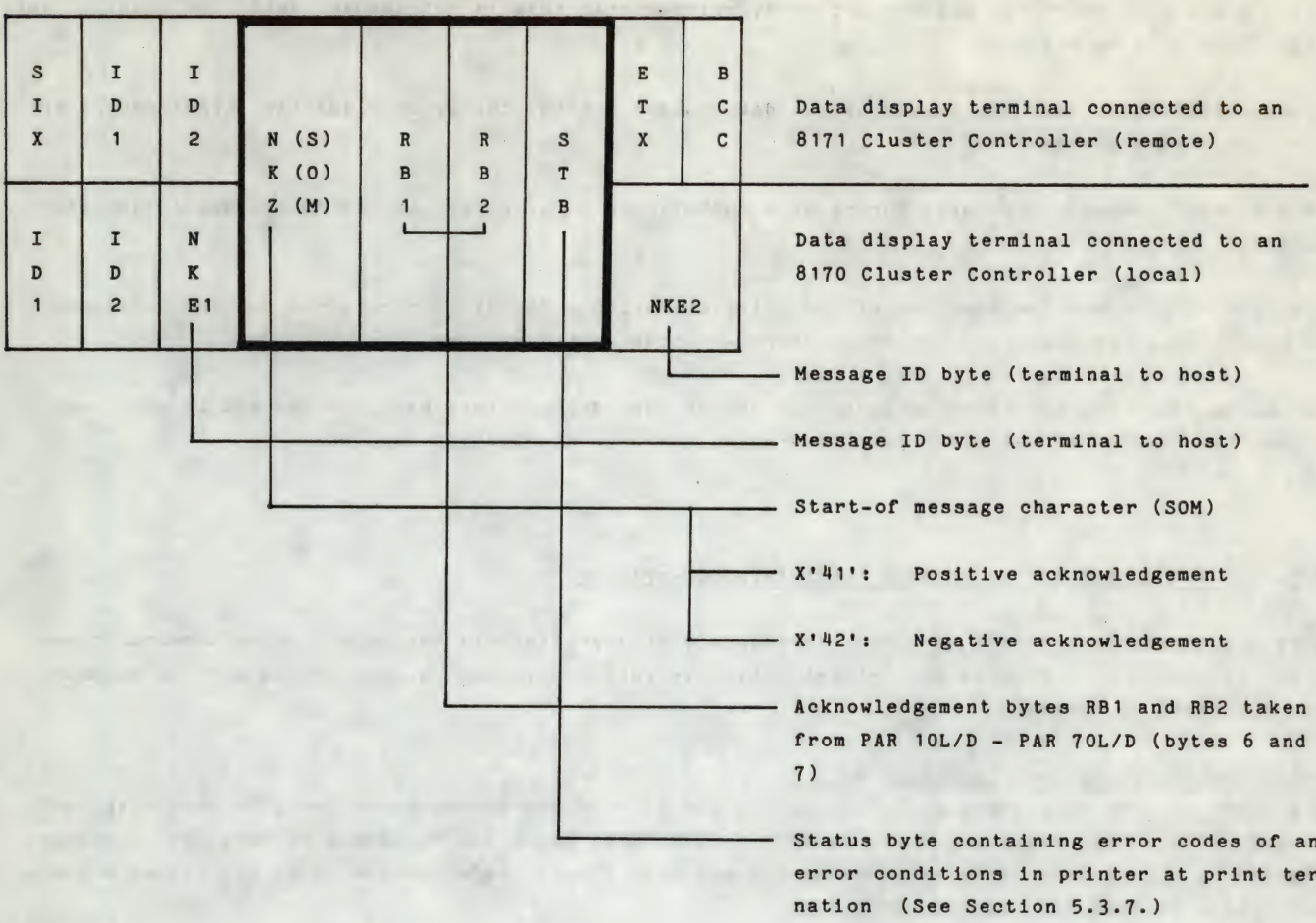
The 9750 Data Display Terminal can handle only one acknowledgement per LA function; and it must be ensured that, if an acknowledgement is requested, no new host-to-terminal transmission using the same parameter range takes place. Acknowledgements are processed serially.

A send request set by means of an acknowledgement is not displayed in the status line and can be reset neither by means of the RS function nor by output from the host.

The acknowledgement request in character PBH (for printers) or ADA (for printer terminals) in parameter range PAR 10L/D - PAR 70L/D is cleared after the job has been performed. In the event of transmission abort, the acknowledgement is lost.

If acknowledgements are not used, the host cannot check whether messages have actually been sent from the data display terminal to the printer.

Message format incorporating an acknowledgement




The acknowledgement comprises the SOM character, acknowledgement bytes RB1 and RB2, and the status byte.

## 5.3 Output to a local printer

### 5.3.1 Readiness polling

Before the actual output from host begins, a printer connected locally to the data display terminal can be polled for readiness via character ADE or AD2. If the printer is not ready, output from host cannot be properly initiated.

In this instance,
either the output is aborted after the addressing sequence (cluster controller mode)
or the selecting sequence is rejected (MSV1 or LSV1 protocol).

Provided the device is fitted with a settings memory (ESP), the type of rejection (WABT or NAK) can be selected (switch S6). The data display terminal's status line does not display an error condition. Polling sequences are also rejected if the data stored in the 9750's send buffer have not yet been processed.

### 5.3.2 Data output via screen

If bits 6 and 5 of character AD2 are set at 0, printer readiness is not checked until the start of data output (from 9750 to printer).

If the printer is already inoperable before data output starts, the error flags LAx, LGERx and Fx are set in the 9750's status line.

If the printer becomes inoperable during data output, the error flags LAx and Fx are set in addition to LGERx.

LGERx is displayed for the duration of the print operation. The 9750 rejects new polling sequences with WABT. This reaction also occurs in shared-printer configurations.

If a shared local printer is occupied by the second data display terminal, the request to print and the LAx flag remain set until the print operation has been started and completed.

### 5.3.3 Data output in bypass mode (host-terminal-printer)

Bypass mode designates a mode of data interchange with a printer via the data display terminal's send buffer, which can be initiated by the host only. In this mode, text is sent to the printer without being displayed on screen.

Printer control characters: interpretation

The printer control characters ESC6 DZ, ESC7 DZ and ESC8 DZ are converted by the 9750 into DZxLF and DZxSPACE (ESC6 = LINE FEED, paper web 1; ESC7 = LINE FEED, paper web 2; ESC8 = SPACE). The distance character DZ may be assigned a value between 1 and 96. (Please refer to the table for distance character values in Appendix II, Section 4.4.)

NB: After printout, the text transferred in bypass mode is no longer available in the data display terminal, i.e. it will have been erased.

If BYPASS is entered in PAR 00D during a data output operation, the message before the parameter entry will be processed normally and displayed on screen. Only the text following PAR 00D, plus any subsequent parameter entries, will be printed out.

The device function entered in PAR 00D is implemented. If this function is an LA function or a P function which itself contains a LA function, the corresponding PAR 10D through 70D will be interpreted. This range must be prepared by the host. Preparation can take place in the same output sequence if PAR 10D through 70D comes before PAR 00D in the message.

NB: Only one message destined for the printer should be in the buffer at any one time, otherwise acknowledgements should be used.

## 5.3.4 Parameter range PAR 10L/D through PAR 70L/D (local printing)

| Character | Designation | Meaning |
|---|---|---|
| 1 | KAN | Channel address<br>00: Local printer connected to data display terminal |
| 2 | PBH | Parameter handling, e.g. acknowledgements |
| 3 | GAD | Device address |
| 4 | GBF | Device command (write-only) |
| 5 | ADA | Print command |
| 6 | RB1 | Acknowledgement byte 1 |
| 7 | RB2 | Acknowledgement byte 2 |
| 8 | | Reserved |

Format and significance of characters in PAR 10L/D through PAR 70L/D (output to local printer)

| Character | Designation | Meaning | Coding | Command | Remarks | Send from |
|---|---|---|---|---|---|---|
| 1 | KAN | Channel address | bit<br>7 . 5 . 3 . 1<br>0 0 0 0 0 0 0 0 | KAN = X'00'<br>Address of a printer connected locally to 9750 | | Host to 9750 to printer |

Format and significance of characters in PAR 10L/D through PAR 70L/D (output to local printer) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks | Send from |
|---|---|---|---|---|---|---|
| 2 | PBH | Parameter handling | bit 7 . 5 . 3 . 1<br>bit 1+6: 0 0 — No acknowledgement to host<br>bit 1+6: 0 1<br>bit 1+6: 1 0 — Pos. and neg. ack.<br>bit 1+6: 1 1 — Neg. ack. only<br>bit 2: 0 — Reserved<br>bit 3: 0<br>1<br>bit 4: =1<br>bit 5: =0 — 9750 outputs 'CR' 'LF' to printer<br>bit 5: =1 — 9750 outputs only 'LF' to printer | | Bit 5 interpreted only for hardcopy (controlled by ADA or 9750 end of line) | 9750 to host<br>Host to 9750 to printer |
| 3 | GAD | Device addresses | X'30' Keyboard<br>X'31' Local dev., GAD1<br>X'32' " " GAD2<br>X'33' " " GAD3<br>X'34' " " GAD4<br>X'35' " " GAD5<br>X'36' " " GAD6<br>X'37' " " GAD7 | Bits 4 and 7 | Via device address, up to 8 devices locally connected to 9750 (incl. keyboard) can be addressed. | Host to 9750 to printer |
| 4 | GBF | Device command | X'53' | Bits 3, 4 and 6 reserved<br>Write command (S) | Write-only operation for printer | |

Function and significance of characters in PAR 10L/D through PAR 70L/D (output to local printer) (cont.)

| Character | Designation | Meaning | Coding | Command | Remarks | Send from |
|---|---|---|---|---|---|---|
| 5 | ADA | Print command | 7 . 5 . 3 . 1<br>bit 1<br>= 0: Hard-copy pr.<br>bit 1<br>1 0 = 1: Forms print<br>bit 2<br>= 0: Output from 9750 without ESC chars.<br>bit 2<br>= 1: 9750 adds ESC3 and ESC4 to start of protected and unprotected field respectively. | | Hard-copy printing in screen format with no contr. chars.<br>Forms printing with all contr. chars. in text | 9750 to printer |
| | | | bit 3,4,5<br>0 0 0 = 80 chars.<br>0 0 1 = 10 "<br>0 1 0 = 32 "<br>0 1 1 = 40 "<br>1 0 0 = 64 "<br>1 0 1 = 80 "<br>1 1 0 = 80 "<br>1 1 1 = 80 " | Bits 3, 4 and 5 for autom. line feed after: | In hard-copy mode (bit 1 of char.5 = 0), autom. line feed in accordance with configuration of bits 3, 4 and 5. Linkable, in PBH char. with carriage return. | |
| 6 | RB1 | Acknowl. byte 1 | X'20' - X'7F' | The serial nos. entered by the host in this char. are sent back unmodified to the host if an acknowledgement is requested. | | Host to 9750 to printer |
| 7 | RB2 | Acknowl. byte 2 | | | | |
| 8 | * | Reserved | | | | |

### 5.3.5 Error messages (local printer)

| Message on 9750 | Displayed | Reset |
|---|---|---|
| GESPERRT (Inhibited) | If the host attempts to modify a protected PAR range by means of the PAR key. | Via RS function. |
| GAD? (Device address?) | If an non-interpretable device address is entered in the PAR range. | |
| PAR-EING (Parameter entry) | If a non-interpretable command code ($\neq$ S) is entered in the PAR range. | |

### 5.3.6 Fx error codes (local printer)

| Error message | From printer |
|---|---|
| F0 | Errored response, e.g.<br>• printer not connected;<br>• incorrect device address set in printer;<br>• incorrect device address entered in PAR 10L/D through PAR 70L/D |
| F1 | Cover open; device error |
| F2 | End of paper; ink supply run out (9002 Printer only) |
| F3 | Errors F1 + F2 |
| F8 | Format error |
| F9 | Errors F1 + F8 |
| FA | Errors F2 + F8 |
| FB | Errors F1 + F2 + F8 |
| FF | I/O channel controller in 9750 faulty or not installed |

An Fx error message can only appear after a print operation has been initiated.

## 5.3.7 Acknowledgements (local printer)

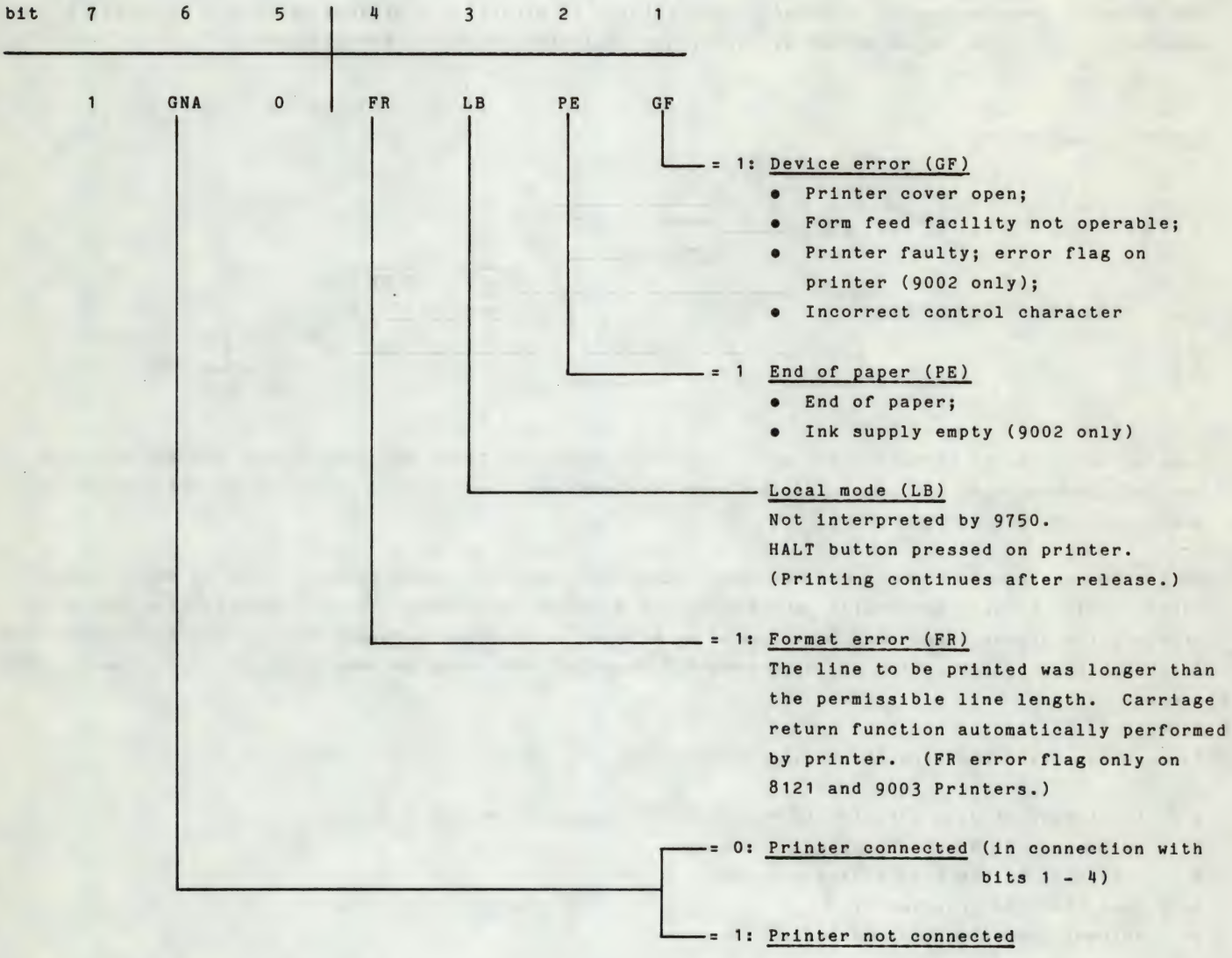
| | | | |
|---|---|---|---|
| S (N)<br>O (B)<br>M (Z) | R<br>B (Ack. byte 1)<br>1 | R<br>B (Ack. byte 2)<br>2 | S<br>T<br>B |

Status byte

**Positive acknowledgement:** In the case of a positive acknowledgement, the status byte is X'40' or X'48'

**Negative acknowledgement:** In the case of a negative acknowledgement, the status byte indicates why the printer is inoperable.

**Status byte format, negative acknowledgement**




A combined error message, e.g. GF + PE, is possible.

## 5.4 Intra-system data interchange with a printer terminal

This mode is valid for transmission sequences from host to cluster controller to printer terminal and vice versa.

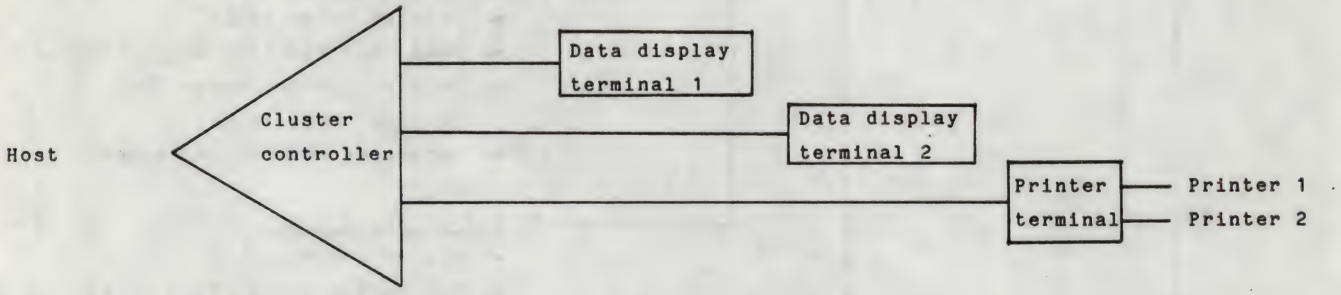
When the LA function is initiated, the message is sent to the printer terminal hooked up to the same cluster controller, provided that the requisite parameter range (PAR 10L/D - PAR 70L/D) has been appropriately assigned. The start-of-message character X'4C' is set by the 9750.

### Message format

| SOM (NBZ) (X'4C') | PAR (8 bytes) (PAR 10L/D - PAR 70L/D) | Text |
|---|---|---|

The parameter entries are interpreted in the printer terminal. The message to be sent - single-block mode only - is delimited on screen by the cursor position and end marker position.

### System configuration




Several data display terminals may access a shared printer. Thus, all 9750's must contain the same channel address entry. It must also be specified which of the printers connected to the printer terminal is to be addressed.

This internal (local) traffic can be inhibited by the host via an appropriate entry in char. PBH of PARAM 1 (8112 Printer Terminal), thus preventing a host-to-9750 send sequence comprising a number of transmission blocks from being interrupted by a local send sequence from a 9750 to the printer terminal (with subsequent destruction of print image). Disabled status can be revoked in the last transmission block.

Internal traffic comprises the following phases:

- initiation of data transfer to 8112 Printer Terminal, including MINZL (minimum cell size) checking;
- buffering in the 8112's data storage;
- data transfer to printer;
- acknowledgements (optional).

Minimum cell size (MINZL)

A specific memory size value - MINZL - must be specified for each of the printers. It is switch-set - by DP maintenance personnel - in increments of 224 bytes up to a maximum of 1792 bytes.

NB
The length of the longest message must be ≤ MINZL. Messages longer than MINZL or longer than the maximum data storage capacity available will cause the connection to be broken. Therefore, for large data columes it is recommended to set MINZL at its maximum value (1792 bytes on the 8112 Printer Terminal). See example in Section 5.4.2.

## 5.4.1 Initiation of data transfer to 8112 Printer Terminal

The LA function initiates data transfer to the 8112 Printer Terminal. Before accepting data, the 8112 checks that the addressed printer is operable and that no internal-traffic inhibit has been set in PARAM 1. The unoccupied data storage space must be equal to or greater than the minimum cell size (MINZL) set.

If the addressed printer is inoperable or disabled or the unoccupied data storage space is less than MINZL, the send request remains set at the 9750 Data Display Terminal. As long as a send request remains set at a data display terminal that wishes to initiate local transfer to a printer terminal, polling of the data display terminal will be rejected with WABT.

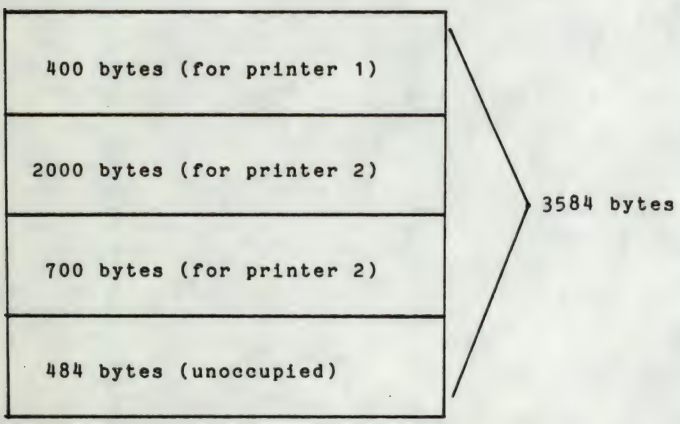
A set send request can be reset by the user via the RS key.

If the addressed printer is operable and enabled and the unoccupied data storage space is equal to or greater than MINZL, data buffering to the 8112's data storage is initiated.

## 5.4.2 Data buffering in 8112 Printer Terminal

If the unoccupied data storage space is equal to or greater than the length of the message, the message will be buffered in its entirety, irrespective of MINZL, and then printed out. If the unoccupied data storage space is less than the length of the message, a negative acknowledgement will be issued and and data transfer retry will be initiated. (The number of retries can be set in the cluster controller.) If enough data storage space becomes free during one of these retries, e.g. after another message has been printed out, the message will be buffered in its entirety and then printed out. If insufficient data storage space becomes free during a retry, the connection will be broken. Manual, i.e. user-initiated, retry will then be necessary.

Examples




MINZL
Printer 1: 1792 bytes
Printer 2: 448 bytes

| Data output | Response of 8112 Printer Terminal |
|---|---|
| Message 1 for printer 1 (400 bytes) | Message 1 is buffered because the unoccupied data storage spece is greater than MINZL. |
| Message 2 for printer 2 (2000 bytes) | Message 2 is buffered because sufficient unoccupied data storage space is available irrespective of MINZL. |
| Message 3 for printer 2 (700 bytes) | Message 3 is buffered because sufficient unoccupied data storage space is available irrespective of MINZL. |
| Message 4 for printer 1 (400 bytes) | Message 4 is not buffered, despite sufficient space being available, because the unoccupied data storage space is less than the MINZL value set. |

### 5.4.3 Data transfer to specified printer

After the data have been buffered error-free in the 8112's data storage, data transfer to the printer in question begins. An acknowledgement for the message, if required, will be generated by the 8112.

NB

Messages transferred to the 8112 are retained in data storage until printout. If one printer goes down, the data storage areas containing messages for this output device will be reconfigured to permit buffering of new messages for the other printer. Negative acknowledgements will be generated for messages lost as a result of this.

## 5.4.4 PAR 10L/D through PAR 70L/D (internal traffic)

| Character | Designation | Meaning |
|---|---|---|
| 1 | KAN | Channel address<br>≠0: Printer terminal connected to cluster controller (internal - local - traffic) |
| 2 | PBH | Parameter handling |
| 3 | GAD | Device address |
| 4 | GBF | Device command (write-only) |
| 5 | ADA | Print command, e.g. acknowledgement |
| 6 | RB1 | Acknowledgement byte 1 |
| 7 | RB2 | Acknowledgement byte 2 |
| 8 | | Reserved |

Format and significance of characters in PAR 10L/D through PAR 70L/D (local output to printer terminal)

| Char. | Desig. | Meaning | Coding | Command | Remarks | Send from |
|---|---|---|---|---|---|---|
| 1 | KAN | Channel address | bit<br>7 . 5 . 3 . 1<br>0 1 — Printer 1 (of pr.term.)<br>1 0 — Printer 2 (of pr.term.)<br>1 1 — Printer 1 or 2 (selected by printer term.) | Chann. address of pr. term. connected to local/remote cluster contr. (message receiver)<br>Printer selection | KAN ≠ X'00' Adress of a pr. term. connected to cluster contr. | Host to cluster contr.to 9750 (to printer term.)to printer term. |

Format and significance of characters in PAR 10L/D through PAR 70L/D (local output to printer terminal) (cont.)

| Char. | Desig. | Meaning | Coding | Command | Remarks | Send from |
|---|---|---|---|---|---|---|
| 2 | PBH | Parameter handling | bit 7 . 5 . 3 . 1<br>bit 1+6: 0 0 = No acknow. to host; 0 1 = Pos. and neg. ack.; 1 0 = Neg. ack. only; 1 1 = Reserved<br>bit 2 = 0: Reserved<br>bit 3 = 0: Reserved<br>1<br>bit 4 = 1:<br>bit 5 = Not interpreted | | In the case of internal traffic, i.e. 9750 to pr. term., PBH is not interpreted in printer terminal. | Host to cluster contr.to 9750 to cluster contr.to printer terminal<br>9750 to cluster contr.to printer terminal |
| 3 | GAD | Device address | bit 7 . 5 . 3 . 1<br>0 1 1 0 1 1 | Bits 4 and 7 reserved.<br>=0: Shared printer, GAD6<br>=1: Shared printer, GAD7 | | 9750 to cluster contr.to printer |
| 4 | GBF | Device command | X'53' | Bits 3, 4 and 6 reserved<br>(write-only) | | |

Format and significance of characters in PAR 10L/D through PAR 70L/D (local output to printer terminal) (cont.)

| Char. | Desig. | Meaning | Coding | Command | Remarks | Send from |
|---|---|---|---|---|---|---|
| 5 | ADA | Print command | bit<br>7 . 5 . 3 . 1<br>1 0<br>0 0 1<br>0 1 0<br>0 1 1<br>1 0 0<br>1 0 1<br>1 1 0<br>1 1 1 | =0: Hard copy<br>=1: Forms printing + acknowledgement<br>10 chars. per line<br>32 " " "<br>40 " " "<br>64 " " "<br>80 " " "<br>120 " " "<br>132 " " "<br>=0: No acknowledgement requested after printout.<br>=1: Acknowledgement requested after printout. | | 9750 to cluster contr.to printer terminal |
| 6 | RB1 | Acknow. byte 1 | X'20' - X'7F' | The serial nos. entered in this char. by the host are returned to the host unmodified if an acknowledgement is requested. | | Host to cl.contr to 9750 to cl. contr.to pr.term. to cl. contr.to 9750 to host |
| 7 | RB2 | Acknow. byte 2 | | | | |
| 8 | * | Reserved | | | | |

Channel address values (cluster-controller-connected printer terminal)

| | EBCDIC | ASCII | EBCDIC | ASCII | EBCDIC | ASCII |
|---|---|---|---|---|---|---|
| Channel | Printer 1 | | Printer 2 | | Free printer | |
| 0 | 40 | 20 | 7C | 40 | 4A | 60 |
| 1 | 5A | 21 | C1 | 41 | 81 | 61 |
| 2 | 7F | 22 | C2 | 42 | 82 | 62 |
| 3 | 7B | 23 | C3 | 43 | 83 | 63 |
| 4 | 5B | 24 | C4 | 44 | 84 | 64 |
| 5 | 6C | 25 | C5 | 45 | 85 | 65 |
| 6 | 50 | 26 | C6 | 46 | 86 | 66 |
| 7 | 7D | 27 | C7 | 47 | 87 | 67 |
| 8 | 4D | 28 | C8 | 48 | 88 | 68 |
| 9 | 5D | 29 | C9 | 49 | 89 | 69 |
| 10 | 5C | 2A | D1 | 4A | 91 | 6A |
| 11 | 4E | 2B | D2 | 4B | 92 | 6B |
| 12 | 6B | 2C | D3 | 4C | 93 | 6C |
| 13 | 60 | 2D | D4 | 4D | 94 | 6D |
| 14 | 4B | 2E | D5 | 4E | 95 | 6E |
| 15 | 61 | 2F | D6 | 4F | 96 | 6F |
| 16 | F0 | 30 | D7 | 50 | 97 | 70 |
| 17 | F1 | 31 | D8 | 51 | 98 | 71 |
| 18 | F2 | 32 | D9 | 52 | 99 | 72 |
| 19 | F3 | 33 | E2 | 53 | A2 | 73 |
| 20 | F4 | 34 | E3 | 54 | A3 | 74 |
| 21 | F5 | 35 | E4 | 55 | A4 | 75 |
| 22 | F6 | 36 | E5 | 56 | A5 | 76 |
| 23 | F7 | 37 | E6 | 57 | A6 | 77 |
| 24 | F8 | 38 | E7 | 58 | A7 | 78 |
| 25 | F9 | 39 | E8 | 59 | A8 | 79 |
| 26 | 7A | 3A | E9 | 5A | A9 | 7A |
| 27 | 5E | 3B | BB | 5B | FB | 7B |
| 28 | 4C | 3C | BC | 5C | 4F | 7C |
| 29 | 7E | 3D | BD | 5D | FD | 7D |
| 30 | 6E | 3E | 6A | 5E | FF | 7E |
| 31 | 6F | 3F | 6D | 5F | 07 | 7F |
| Local printer connected to 9750 | | | | | 00 | 00 |

## 5.4.5 Acknowledgement format

| NBZ (SOM) | RB1, e.g. program no. (user) | RB2, e.g. message serial no. | Status byte |
|---|---|---|---|

| Character | Designation | Coding | Meaning | |
|---|---|---|---|---|
| 1 | NBZ | X'41'<br>X'42' | Positive acknowledgement<br>Negative acknowledgement | |
| 2 | RB1 | X'20' - X'7F' | Acknowledgement byte 1 | RB1 and RB2 in PAR 10L/D - 70L/D are returned unmodified to host. |
| 3 | RB2 | X'20' - X'7F' | Acknowledgement byte 2 | |
| 4 | STB | X'41' - X'54' | Status byte (assignment for printer) | |

Status byte format, positive acknowledgement

| Bit | 7 | 6 | 5 | 4 | 3 | 2 | 1 | |
|---|---|---|---|---|---|---|---|---|
| | 1 | 0 | 0 | 0 | 0 | 0 | 0 | X'40': Message printed in its entirety. |
| | 1 | 0 | 0 | 1 | 1 | 0 | 1 | X'48': Message printed in its entirety, but with format error(s). |

As the 8112 does not issue a logical acknowledgement until a message has been printed out completely, a negative acknowledgement in this instance is not possible.

# 6 Operations using the badge reader

The badge reader integrated into the 9750 Data Display Terminal's keyboard permits files protected by the operating system or by user programs to be accessed on the basis of a user ID entered by means of a badge.

## 6.1 Badge formats

The badge reader reads SIPASS and ABA-format badges. The reader is factory-set for SIPASS format, but can be adjusted via a jumper in the keyboard. (Please refer to the relevant setup and adjustment instructions.)

## 6.2 Entering badge information

In order to permit badge information to be read and transferred, the request bit must be set in character FST8 of PAR 01L by the host - see Section 2.4.6 for coding - and an enter request must be displayed on screen. The format and representation of the enter request are specified by the operating system or the user program.

Once the request bit is set in PAR 01L, keyboard entry is inhibited. A read attempt without a prior request from the host will cause the message GESPERRT (Disabled) to be displayed in line 25 of the screen.

Bit 1 of character ZZ1 in PAR 00E is set to 1 when a valid item of badge information is entered. At the same time, the data display terminal resets the request bit in PAR 01L and the keyboard is released. Bit 1 of character ZZ1 - see Section 2.4.10 for PAR 00E - remains set until the badge has been removed from the reader. After the badge has been removed, the reader sends a K14 message to the host.

## 6.3 Entering errored badge information

If there is no badge or if the badge information is entered incorrectly, the read poll from the host must be responded to by means of a K14 message via the keyboard. By outputting a message, the operating system or user program can now reset the request bit in PAR 01L, thus releasing the keyboard. However, access to protected files is not possible.

## 6.4 Sending a K14 message: keyboard initiation

The K14 must be sent via an ESC sequence, i.e. by actuating the keys ESC and Colon.

## 6.5 Error messages

The badge reader itself does not contain an error flag facility. Error messages are displayed on the screen of the 9750 Data Display Terminal. The following table contains error message texts, information on probable causes, and suggested error recovery measures.

| Error message text | Cause of error | Error recovery measures |
|---|---|---|
| AUSWEISFEHLER<br>(Badge error) | • Badge passed through reader too slowly, at non-constant speed, or at an angle. | • Repeat read operation. |
| | • Badge is defective or badge information is invalid. | • Replace badge. |
| | • Badge reader set to incorrect format. | • Adjust format setting |
| KARTE LEER<br>(Badge empty) | • Virgin - blank - badge. | • Replace badge. |
| | • Badge passed through reader at incorrect attitude. | • Repeat read operation, observing correct attitude. |
| GESPERRT<br>(Disabled)<br>(in status line) | • Read attempt without prior request from host. | • Actuate RS key. |

# Appendix I Operations with BERMUDA (User Service for Terminal Mask Support)

## 1 BERMUDA

BERMUDA, the User Service for Terminal Mask Support, is a component of TRANSDATA PDN (Program System for Teleprocessing and Network Control) and is a high-convenience tool for supporting data terminals connected to terminal computers from the Siemens TRANSDATA 960 Communication Computer System.

The principal functional features of BERMUDA are:

- input support for formatted data,
- formatted data output,
- editing for printout of formatted data.

Input support for formatted data is geared essentially to high-speed source data entry for storage in LAM (Library Access Method) files. The terminal computer outputs a mask to the terminal's screen. The mask contains operator prompt texts (in protected fields), variable fields to be filled in by the user, and an error message line. (See Section 1.4 of this Appendix.)

In field mode (see Section 1.1 below), the terminal computer also outputs rules governing the filling-in of the masks (field specifications).

The 9750 Data Display Terminal continues to be responsible for checking that the syntax of input data is in line with current field specifications.

Once entered, input data can be subjected to keyboard crosschecking, i.e. they are checked by being reentered.

### 1.1 Operating modes

The 9750 Data Display Terminal can use BERMUDA in record mode and field mode. The terminal computer activates the modes by loading the parameter ranges PAR 00L and PAR 02L. After power-up, the 9750 is always set to record mode. (The user cannot perform mode switchover.)

#### 1.1.1 Record mode

Record mode is intended for video terminals that are not field-mode-capable, e.g. the 8160 Data Display Terminal.

The terminal computer outputs a mask to screen. The mask contains operator prompts, as well as variable fields to be filled in by the user. Line 24 is the BERMUDA indicator line, line 25 the device status line.

The user can fill in all variable fields in the mask successively; any position can be occupied. After the mask has been partly or completely filled in, the user can initiate a send sequence to the terminal computer. BERMUDA then takes over the processsing of the input data, i.e. syntax checking and field editing.

### 1.1.2 Field mode

In field mode, BERMUDA releases only one mask field - designated the current field - at a time for filling in by the user. The cursor is set to the first position of the current input field. The 9750 performs a syntax check for every character entered. Transmission is initiated only after this check is successfully completed: the field is edited and sent to the terminal computer, which, in its next output sequence, releases the next field for filling in by the user.

## 1.2 Parameter range overview

| | Parameter ranges for the user<br>Keys :<br>DÜ 1) | | | | | | | | Parameter ranges for host-controlled functions<br>Functions:<br>LA1 | LA2 | LA3 | LA4 | LA5 | LA6 | LA7 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADDRESS OF PARAMETER SECTION | ADDRESS OF PARAMETER RANGE<br>X'40' | X'41' | X'42' | X'43' | X'44' | X'45' | X'46' | X'47' | X'48' | X'49' | X'4A' | X'4B' | X'4C' | X'4D' | X'4E' | X'4F' |
| X'40' | PAR00L | PAR10L | PAR20L | PAR30L | PAR40L | PAR50L | PAR60L | PAR70L | PAR00D | PAR10D | PAR20D | PAR30D | PAR40D | PAR50D | PAR60D | PAR70D |
| X'41' | PAR01L | PAR11L | PAR21L | PAR31L | PAR41L | PAR51L | PAR61L | PAR71L | 2)<br>PAR01D | PAR11D | PAR21D | PAR31D | PAR41D | PAR51D | PAR61D | PAR71D |
| X'42' | PAR02L | | | | | | | | 2)<br>PAR02D | | | | | | | |
| X'43' | | | | | | | | | 2)<br>PAR03D | | | | | | | |
| X'40' | BERMUDA INDICATOR (STATUS) LINE | | | | | | | | | | | | | | | |

For reasons of compatibility, output of a message header in the form hitherto used causes parameter ranges PARAM0, PARAM1 and PARAM2 to be entered in PAR 00L and PAR 00D, PAR 10L and PAR 10D, and PAR 20L and PAR 20D respectively.

1) All DÜ, F and K keys.

2) Mailbox facility for storing a text ID.

## 1.2.1 Parameter ranges for host-to-terminal transmission

### 1.2.1.1 PAR 00L

| Character | Designation | Coding | Command |
|---|---|---|---|
| 1 | SAW1 | X'47' | Send current field only, i.e. the field containing the cursor. 1) |
| 2 | FST1 | X'49' | Keyboard initiation of EFZ and AFZ functions inhibited. Cursor can be positioned in protected fields. |
| 3 | FST2 | X'79' | Modify and flash bits (MOD and BLINKEN) are erased in unprotected fields from start of screen to end of screen. If the LZF function is entered, these bits these bits are unaffected. |
| 4 | GEF1 | X'00' | All device functions are controlled by parameter PAR 00D. |
| 5 | FST3 | X'5C' | Dialog mode activated. Cursor remains stationary. Message header for terminal-to-host transmission comprises PAR 00E + PAR 01E. |
| 6 | FST4 | X'42' | Field mode activated. |
| 7 | GEF2 | X'00' | |
| 8 | WAR | X'44' | Sent unmodified in character 8 (WARÜ in PAR 00E) in next transmission sequence from terminal to host. |

Note: 1) In field mode with or without keyboard crosschecking, SAW X'47' must be used. Every time a send sequence is initiated, PARx0E defines the field containing the cursor as the "current field" and transmits it to the host. The host may not enter a start marker or a logical end of line in the "current field". (In field mode, they cannot be entered via the keyboard.)

Character 5 (AZL) in PAR 00D:
bit 1 = 0: Status (indicator) line displayed
bit 1 = 1: System line displayed

(See also Section 1.4 of this Appendix.)

### 1.2.1.2 PAR 02L

| Character | Designation | Coding | Command |
|---|---|---|---|
| 1 | -- | X'40' | -- |

| Character | Designation | Coding | Command |
|---|---|---|---|
| 2 | FVG1<br>Field specif. 1 | bit<br>7 . 5 . 3 . 1<br>1 | |
| | | bit 1 — 0: | Upper-case letters and blanks permitted. |
| | | bit 1 — 1: | Blanks and all upper-case letters except A, B, C, D, E and F blocked. Upper-case A through F are also used as hex numbers and are only blocked if hex numbers are blocked, i.e. if bit 6 is 1. |
| | | bit 2 — 0: | Decimal numbers permitted. |
| | | bit 2 — 1: | Not permitted; only effective if bit 6 is 1. |
| | | bit 3 — 0: | Class 1 special characters permitted, i.e. ! " # $ % & ' ( - = ? / ; : * @ |
| | | bit 3 — 1: | Not permitted. |
| | | bit 4 — 0: | Period and comma permitted. |
| | | bit 4 — 1: | Not permitted. |
| | | bit 5 — 0: | Plus and minus signs permitted. |
| | | bit 5 — 1: | Not permitted. |
| | | bit 6 — 0: | Hex numbers - 0 through F - permitted. |
| | | bit 6 — 1: | Not permitted. |
| 3 | FVG2<br>Field specif. 2 | bit<br>7 . 5 . 3 . 1<br>1 | |
| | | bit 1 — 0: | Lower-case letters permitted. |
| | | bit 1 — 1: | Not permitted. |
| | | bit 2 — 0: | Fill field with blanks. |
| | | bit 2 — 1: | Fill field with null chars. |
| | | bit 3 — 0: | Shift field contents and right-justify. |
| | | bit 3 — 1: | Do not shift. |
| | | bit 4 — 0: | Not a mandatory field. |
| | | bit 4 — 1: | Mandatory field. |

| Character | Designation | Coding | Command |
|---|---|---|---|
| 3 | FVG2 | bit 5 | 0: Not an integrity field. |
| | | bit 5 | 1: Integrity field. |
| | | bit 6 | 0: Class 2 special characters permitted, i.e. upper and lower-case umlaut characters, e.g. Ä, ä, Ö, ö, plus ß. |
| | | bit 6 | 1: Not permitted. |
| 4 | Field lengths LH | X'30' - X'39' | |
| 5 | LZ | | Decimal place of entire field length. |
| 6 | LE | | |
| 7 | FKZ Function ID | bit 7 . 5 . 3 . 1 | |
| | | bit 1 | 0: No keyboard crosschecking. |
| | | 1 0 bit 1 | 1: Keyboard crosschecking. |
| | | bit 2 | 0: KOR permissible. |
| | | bit 2 | 1: KOR not permissible. |
| 8 | -- | X'40' | No meaning assigned. |

## 1.2.2 Parameter ranges for terminal-to-host transmission

In transmissions from terminal to host in field mode, the message is preceded by parameter ranges PAR 00E, PAR 01E and PAR 02E.

Message format for terminal-to-host transmission in field mode:

```
Length              1 / 1 / 1 /   8    / 8   /   8    / ... 0 through n ...
Meaning           ID1/ID2/NBZ/PAR 00E/PAR 01E/PAR 02E/ ... text ...

                        SOM
```

### 1.2.2.1 PAR 00E

The parameter range PAR 00E in field mode corresponds to the same parameter range in record mode except for the following:

(1) in field mode, code X'00' is entered in character CDS;
(2) the code of the function initiating a send operation is entered in parameter PAR 02E.

### 1.2.2.2 PAR 01E

The parameter range PAR 01E in field mode corresponds to the same parameter range in block mode.

### 1.2.2.3 PAR 02E

In field mode, the 9750 provides information on the type of entry for every entry in parameter range PAR 02E.

| Character | Designation | Coding | Command |
|---|---|---|---|
| 1 | ZZ3 Status Character 3 | X'40<br>X'41' | VALID not confirmed.<br>VALID confirmed. |
| 2 | STA1 | X'40' - X'7F' | Number of corrections via keyboard cross-checking. |
| 3 | STA2 | X'40'<br>.<br>X'7F' | Statistics: binary representation of number of keystrokes actuated with regard to the field; STA3 contains low-order byte. |
| 4 | STA3 | | |
| 5 | STA4 | X'40' - X'7F' | Number of flashers in field. |
| 6 | -- | X'40' | No meaning assigned. |
| 7 | CDF1 | X'40' - X'7F' | Code of function initiating a send operation. |
| 8 | CDF2 | | |

Please refer to section on data formats for descriptions. The statistics messages are reset at end of output of every message that affects the screen.

## 1.3 Status (indicator) line

Line 25 is normally the BERMUDA indicator line (system line). However, if device errors occur or characters which violate the syntax rules are entered, the BERMUDA indicator line is replaced by the device status line. The BERMUDA indicator line is divided into two areas of 40 characters each. The area on the left is used for displaying field-specific error messages. Outputs in this area are combined with the flasher and an audible signal. The error message texts are explained in Section 1.4 of this Appendix.

## 1.4 Messages in BERMUDA indicator line

| | |
|---|---|
| FKT-TASTE UNZULÄSSIG | The current field has been terminated by means of an invalid function key; it must be terminated with a valid key. |
| POS. NICHT MOEGLICH | One of the field terminating keys ← → has been pressed in a field from which the associated cursor positioning function cannot be implemented. |
| KEIN DUP-SATZ | Manual duplication is not possible because no duplication record has been generated by means of the DUPS key. |
| DUP.-EIN | The DUPS key has been depressed, thereby activating duplication mode, i.e. a copy of the net data record will be generated at the end of the record. |
| DUP.-AUS | Duplication mode has been terminated by means of the DUPS key; no additional copy of a net data record will be produced at the end of the record. |
| FOLGESEGMENT ZU LANG | The next segment of a segmented format is too long to be displayed on screen. |
| BERMUDA TEST EIN | Test mode has been activated by means of the TEST key. |
| BERMUDA TEST AUS | Test mode has been deactivated by means of the AKT key. |
| BERMUDA DIAG EIN | Diagnostic mode has been activated by means of the DIAG key. |
| + | Sign error |
| ; | Decimal point/place error |

## 1.5 Functions of BERMUDA keyboard

The function keys of the BERMUDA-oriented keyboard of a 9750 Data Display Terminal can be divided into two categories:

- local-function keys,
- field-terminating function keys.

### 1.5.1 Local-function keys

Local-function keys trigger functions in the data display terminal, but have no impact on programs executing in the terminal computer.

Actuating the local-function keys - SML, SMR, EFG and AFG - triggers the same functions as in record mode, the only difference being that in field mode a transmission sequence is initiated as soon as a field is left.

### 1.5.2 Field-terminating function keys

The field-terminating function keys on the 9750 Data Display Terminal's keyboard can be subdivided into four categories:

- system function keys,
- user function keys,
- special-function keys,
- invalid function keys.

#### 1.5.2.1 System function keys

These keys permit the terminal user to terminate his input to the current field without having to fill in the field to the last character. When a system function key is pressed, the current field is edited as specified for its type and then passed on to the terminal computer. The data display terminal sends the code of the function initiating transmission to the terminal computer in parameter PAR 02E. Depending on the code, BERMUDA then implements a specific, defined function.

#### 1.5.2.2 User function keys

These function keys permit the terminal user to send project-specific short messages (telegrams) to the application program. Actuating one of these keys does not terminate entry into the current input field. The use of user function keys is not permitted if the current input field is an empty mandatory field or a partially filled integrity field.

#### 1.5.2.3 Special-function keys

Application-dependent.

#### 1.5.2.4 Invalid function keys

No meaningful processing instruction is communicated to BERMUDA if one of the function keys listed below is pressed. Key actuation is rejected with the message FKT-TASTE UNZULAESSIG (invalid function

key) in the BERMUDA indicator line. If an invalid key is pressed, the cursor moves to the start of the current input field.

The following function keys are invalid:

| | |
|---|---|
| SMO | Cursor up |
| SMU | Cursor down |
| SZA | Cursor to start of current line |
| SDZ | Cursor to start of preceding line |
| LZE | Logical end of line |
| VA | Clear connection (applicable only if the 9750 is a standalone terminal) |
| RU | Roll-up |
| MAR | Mark field |
| F4 | |
| F5 | |

The use of the following keys is not permitted in BERMUDA mode. (If depressed, they cause a flashing error flag (FEHL) to be displayed in the device status line.)

| | |
|---|---|
| LSP | Erase memory |
| EFZ | Insert line |
| AFZ | Delete line. |

Resetting is possible by means of the RS key.

## 1.6 Keyboard crosschecking

This permits the original entry to be checked by being reentered. This mode is set by the terminal computer in parameter PAR 02L - see Section 1.2.1.2 of this Appendix - and cannot be activated by the user. The keys ANZ, KOR and VALID are provided specifically for this mode.

### 1.6.1 Crosschecking data entry

The terminal computer displays a mask with original entered data on the 9750's screen; the cursor is positioned at the start of the current field. The entered data appear obscured on screen; the user must reenter the data to check it.

The current field is handled like an integrity and mandatory field, i.e. null characters and underscore may not be entered.

Each character entered is immediately compared in the 9750 with the character output by the terminal computer. If the characters are identical, the entered character is visibly displayed on screen and the cursor jumps to the next scratch character. If the characters are not identical, the flasher is tripped and the error message P appears is displayed in the indicator line.

Once the field has been filled completely and error-free, it is sent to the terminal computer for validity-checking. If the check proves positive, the terminal computer releases the next field; if the check proves negative, the flasher is activated.

### 1.6.2 Additional crosschecking functions

ANZ Display field

Depressing this key causes all data that have not yet been checked to be displayed. After a data character has been entered or a non-transmission function initiated, all unchecked data are again displayed as scratch characters. Pressing the ANZ key has no effect if no valid character has yet been entered.

KOR Correct

If a first-time character sent by the terminal computer is incorrect, it can be corrected by pressing the KOR key and entering the appropriate data character. The corrected character is the next character to be checked. Thus, it must be entered twice.

VALID and LZE

In keyboard crosschecking mode, the LZE key initiates the FKOR function (Field correction). The VALID and FKOR functions can be used to leave keyboard crosschecking mode prematurely.

VALID

Keyboard crosschecking and syntax checking are deactivated by pressing the VALID key. The 9750 is then in field mode. The next output sequence from the terminal computer that affects the screen reactivates keyboard crosschecking mode.

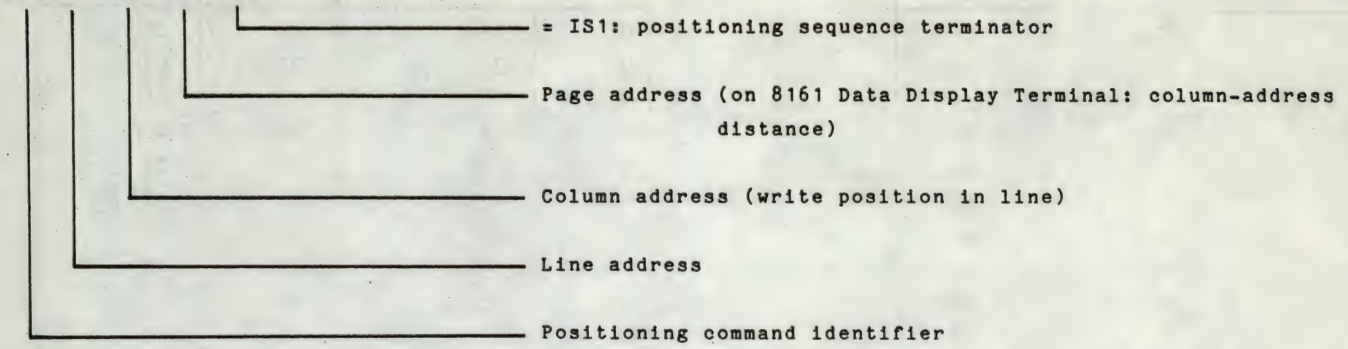
FKOR Field correction

Actuating the LZE key initiates data transmission to the terminal computer and activates the dialog inhibit (DIA).

## Appendix II Output-message format tables

### 1 Absolute cursor positioning commands (IS4 sequences)

Format of control character sequence




### 1.1 Line address values

In the 9750 Data Display Terminal, IS4 sequences of both code types are output without additional adjustments or alterations. (See tables in this Section and in Section 1.2 of this Appendix.)

For transmission from terminal to host, different coding of the IS4 line address can be requested, depending on the status of bit 2 of character 5 in PAR 00L/D: if bit 2 is 0, coding is as per the table in Section 1.2; if bit 2 is 0, coding is as per the table below.

| Line address | EBCDIC | ASCII |
|---|---|---|
| 1 | 07 | 7F |
| 2 | FF | 7E |
| 3 | FD | 7D |
| 4 | 4F | 7C |
| 5 | FB | 7B |
| 6 | A9 | 7A |
| 7 | A8 | 79 |
| 8 | A7 | 78 |
| 9 | A6 | 77 |
| 10 | A5 | 76 |
| 11 | A4 | 75 |
| 12 | A3 | 74 |
| 13 | A2 | 73 |
| 14 | 99 | 72 |
| 15 | 98 | 71 |
| 16 | 97 | 70 |
| 17 | 96 | 6F |
| 18 | 95 | 6E |
| 19 | 94 | 6D |
| 20 | 93 | 6C |
| 21 | 92 | 6B |
| 22 | 91 | 6A |
| 23 | 89 | 69 |
| 24 | 88 | 68 |

| Line addresses sorted by EBCDIC/ASCII code | | | |
|---|---|---|---|
| EBCDIC | Line addr. | ASCII | Line addr. |
| 07 | 1 | 68 | 24 |
| 4F | 4 | 69 | 23 |
| 88 | 24 | 6A | 22 |
| 89 | 23 | 6B | 21 |
| 91 | 22 | 6C | 20 |
| 92 | 21 | 6D | 19 |
| 93 | 20 | 6E | 18 |
| 94 | 19 | 6F | 17 |
| 95 | 18 | 70 | 16 |
| 96 | 17 | 71 | 15 |
| 97 | 16 | 72 | 14 |
| 98 | 15 | 73 | 13 |
| 99 | 14 | 74 | 12 |
| A2 | 13 | 75 | 11 |
| A3 | 12 | 76 | 10 |
| A4 | 11 | 77 | 9 |
| A5 | 10 | 78 | 8 |
| A6 | 9 | 79 | 7 |
| A7 | 8 | 7A | 6 |
| A8 | 7 | 7B | 5 |
| A9 | 6 | 7C | 4 |
| FB | 5 | 7D | 3 |
| FD | 3 | 7E | 2 |
| FF | 2 | 7F | 1 |

## 1.2 Line address values for 8161-compatible mode

| Line address | EBCDIC | ASCII |
|---|---|---|
| 1 | F7 | 37 |
| 2 | F6 | 36 |
| 3 | F5 | 35 |
| 4 | F4 | 34 |
| 5 | F3 | 33 |
| 6 | F2 | 32 |
| 7 | F1 | 31 |
| 8 | F0 | 30 |
| 9 | 61 | 2F |
| 10 | 4B | 2E |
| 11 | 60 | 2D |
| 12 | 6B | 2C |
| 13 | 4E | 2B |
| 14 | 5C | 2A |
| 15 | 5D | 29 |
| 16 | 4D | 28 |
| 17 | 7D | 27 |
| 18 | 50 | 26 |
| 19 | 6C | 25 |
| 20 | 5B | 24 |
| 21 | 7B | 23 |
| 22 | 7F | 22 |
| 23 | 5A | 21 |
| 24 | 40 | 20 |

| Line addresses sorted by EBCDIC and ASCII | | | |
|---|---|---|---|
| EBCDIC | Line addr. | ASCII | Line addr. |
| 40 | 24 | 20 | 24 |
| 4B | 10 | 21 | 23 |
| 4D | 16 | 22 | 22 |
| 4E | 13 | 23 | 21 |
| 50 | 18 | 24 | 20 |
| 5A | 23 | 25 | 19 |
| 5B | 20 | 26 | 18 |
| 5C | 14 | 27 | 17 |
| 50 | 15 | 28 | 16 |
| 60 | 11 | 29 | 15 |
| 61 | 9 | 2A | 14 |
| 6B | 12 | 2B | 13 |
| 6C | 19 | 2C | 12 |
| 7B | 21 | 2D | 11 |
| 7D | 17 | 2E | 10 |
| 7F | 22 | 2F | 9 |
| F0 | 8 | 30 | 8 |
| F1 | 7 | 31 | 7 |
| F2 | 6 | 32 | 6 |
| F3 | 5 | 33 | 5 |
| F4 | 4 | 34 | 4 |
| F5 | 3 | 35 | 3 |
| F6 | 2 | 36 | 2 |
| F7 | 1 | 37 | 1 |

### 1.2.1 Column-address distance values (feed)

| Distance | EBCDIC | ASCII |
|---|---|---|
| 0 | F0 | 30 |
| -1 | F1 | 31 |
| -2 | F2 | 32 |

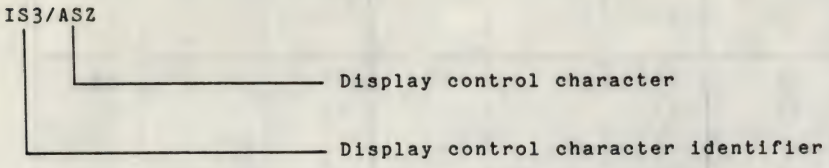
## 1.3 Column address values

| Col. addr. | EBCDIC | ASCII | Col. addr. | EBCDIC | ASCII | Col. addr. | EBCDIC | ASCII | Col. addr. | EBCDIC | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 07 | 7F | 21 | 92 | 6B | 41 | E6 | 57 | 61 | C3 | 43 |
| 2 | FF | 7E | 22 | 91 | 6A | 42 | E5 | 56 | 62 | C2 | 42 |
| 3 | FD | 7D | 23 | 89 | 69 | 43 | E4 | 55 | 63 | C1 | 41 |
| 4 | 4F | 7C | 24 | 88 | 68 | 44 | E3 | 54 | 64 | 7C | 40 |
| 5 | FB | 7B | 25 | 87 | 67 | 45 | E2 | 53 | 65 | 6F | 3F |
| 6 | A9 | 7A | 26 | 86 | 66 | 46 | D9 | 52 | 66 | 6E | 3E |
| 7 | A8 | 79 | 27 | 85 | 65 | 47 | D8 | 51 | 67 | 7E | 3D |
| 8 | A7 | 78 | 28 | 84 | 64 | 48 | D7 | 50 | 68 | 4C | 3C |
| 9 | A6 | 77 | 29 | 83 | 63 | 49 | D6 | 4F | 69 | 5E | 3B |
| 10 | A5 | 76 | 30 | 82 | 62 | 50 | D5 | 4E | 70 | 7A | 3A |
| 11 | A4 | 75 | 31 | 81 | 61 | 51 | D4 | 4D | 71 | F9 | 39 |
| 12 | A3 | 74 | 32 | 4A | 60 | 52 | D3 | 4C | 72 | F8 | 38 |
| 13 | A2 | 73 | 33 | 6D | 5F | 53 | D2 | 4B | 73 | F7 | 37 |
| 14 | 99 | 72 | 34 | 6A | 5E | 54 | D1 | 4A | 74 | F6 | 36 |
| 15 | 98 | 71 | 35 | BD | 5D | 55 | C9 | 49 | 75 | F5 | 35 |
| 16 | 97 | 70 | 36 | BC | 5C | 56 | C8 | 48 | 76 | F4 | 34 |
| 17 | 96 | 6F | 37 | BB | 5B | 57 | C7 | 47 | 77 | F3 | 33 |
| 18 | 95 | 6E | 38 | E9 | 5A | 58 | C6 | 46 | 78 | F2 | 32 |
| 19 | 94 | 6D | 39 | E8 | 59 | 59 | C5 | 45 | 79 | F1 | 31 |
| 20 | 93 | 6C | 40 | E7 | 58 | 60 | C4 | 44 | 80 | F0 | 30 |

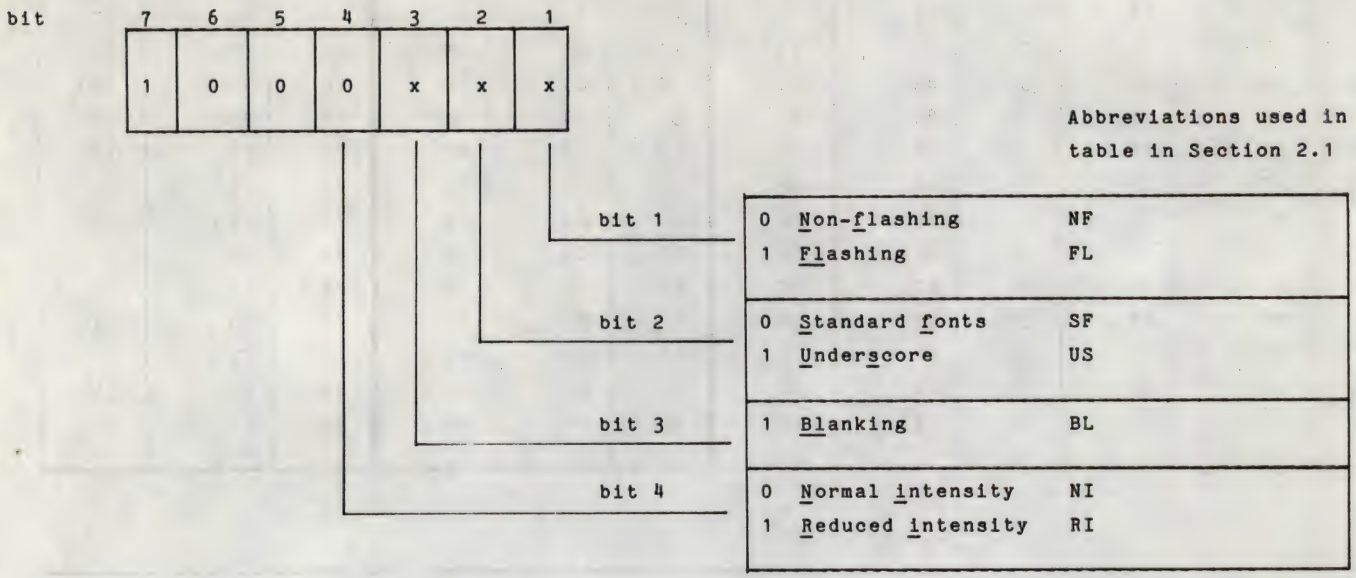
| Column-address values sorted by EBCDIC and | | | | | | | | ASCII | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| EBC DIC | Col. addr. | EBC DIC | Col. addr. | EBC DIC | Col. addr. | EBC DIC | Col. addr. | ASCII | Col. addr. | ASCII | Col. addr. | ASCII | Col. addr. | ASCII | Col. addr. |
| 07 | 1 | 89 | 23 | BD | 35 | E3 | 44 | 30 | 80 | 44 | 60 | 58 | 40 | 6C | 20 |
| 4A | 32 | 91 | 22 | C1 | 63 | E4 | 43 | 31 | 79 | 45 | 59 | 59 | 39 | 6D | 19 |
| 4C | 68 | 92 | 21 | C2 | 62 | E5 | 42 | 32 | 78 | 46 | 58 | 5A | 38 | 6E | 18 |
| 4F | 4 | 93 | 20 | C3 | 61 | E6 | 41 | 33 | 77 | 47 | 57 | 5B | 37 | 6F | 17 |
| 5E | 69 | 94 | 19 | C4 | 60 | E7 | 40 | 34 | 76 | 48 | 56 | 5C | 36 | 70 | 16 |
| 6A | 34 | 95 | 18 | C5 | 59 | E8 | 39 | 35 | 75 | 49 | 55 | 5D | 35 | 71 | 15 |
| 6D | 33 | 96 | 17 | C6 | 58 | E9 | 38 | 36 | 74 | 4A | 54 | 5E | 34 | 72 | 14 |
| 6E | 66 | 97 | 16 | C7 | 57 | F0 | 80 | 37 | 73 | 4B | 53 | 5F | 33 | 73 | 13 |
| 6F | 65 | 98 | 15 | C8 | 56 | F1 | 79 | 38 | 72 | 4C | 52 | 60 | 32 | 74 | 12 |
| 7A | 70 | 99 | 14 | C9 | 55 | F2 | 78 | 39 | 71 | 4D | 51 | 61 | 31 | 75 | 11 |
| 7C | 64 | A2 | 13 | D1 | 54 | F3 | 77 | 3A | 70 | 4E | 50 | 62 | 30 | 76 | 10 |
| 7E | 67 | A3 | 12 | D2 | 53 | F4 | 76 | 3B | 69 | 4F | 49 | 63 | 29 | 77 | 9 |
| 81 | 31 | A4 | 11 | D3 | 52 | F5 | 75 | 3C | 68 | 50 | 48 | 64 | 28 | 78 | 8 |
| 82 | 30 | A5 | 10 | D4 | 51 | F6 | 74 | 3D | 67 | 51 | 47 | 65 | 27 | 79 | 7 |
| 83 | 29 | A6 | 9 | D5 | 50 | F7 | 73 | 3E | 66 | 52 | 46 | 66 | 26 | 7A | 6 |
| 84 | 28 | A7 | 8 | D6 | 49 | F8 | 72 | 3F | 65 | 53 | 45 | 67 | 25 | 7B | 5 |
| 85 | 27 | A8 | 7 | D7 | 48 | F9 | 71 | 40 | 64 | 54 | 44 | 68 | 24 | 7C | 4 |
| 86 | 26 | A9 | 6 | D8 | 47 | FB | 5 | 41 | 63 | 55 | 43 | 69 | 23 | 7D | 3 |
| 87 | 25 | BB | 37 | D9 | 46 | FD | 3 | 42 | 62 | 56 | 42 | 6A | 22 | 7E | 2 |
| 88 | 24 | BC | 36 | E2 | 45 | FF | 2 | 43 | 61 | 57 | 41 | 6B | 21 | 7F | 1 |

## 2 Display control characters for field presentation (IS3 sequences)

Format of control character sequence




Format of display control characters




Abbreviations used in table in Section 2.1

### 2.1 Display control character values

| Attribute | EBCDIC | ASCII |
|---|---|---|
| NI, SF, NF | 7C | 40 |
| NI, SF, FL | C1 | 41 |
| NI, US, NF | C2 | 42 |
| NI, US, FL | C3 | 43 |
| RI, SF, NF | C8 | 48 |
| RI, SF, FL | C9 | 49 |
| RI, US, NF | D1 | 4A |
| RI, US, FL | D2 | 4B |
| BL ** | C4 | 44 |

** Attribute combinations with blanking do not appear practical.

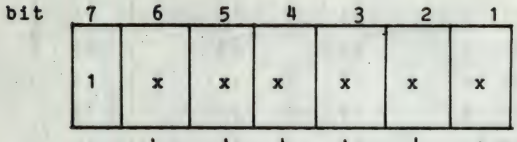
## 3 Field handling characters for field attribute specification (IS2 sequences)

**Format of control character sequence**

IS2/FBZ

- Field handling character
- Field handling character identifier

**Format of field handling character**




| | | Abbreviations used in table in Section 3.1 |
|---|---|---|
| bit 6, 1 | | |
| 00 | Variable fields | V |
| 01 | Protected fields | P |
| 10 | Protected fields, sendable fields | PS |
| 11 | Protected fields; field addresses only are sendable | PA |
| bit 2 | | |
| 0 | Alphanumeric input permitted | A |
| 1 | Numeric input only | N |
| bit 3 | | |
| | Modified bit; not program-controllable | MO |
| bit 4 | | |
| 0 | Non-markable field | NM |
| 1 | Markable field | MA |
| bit 5 | | |
| 0 | Non-printable field | NP |
| 1 | Printable field | PF |

## 3.1 Field-handling character values 1)

| Field attribute | EBCDIC | ASCII | Field attribute | | EBCDIC | ASCII |
|---|---|---|---|---|---|---|
| V, NM, NP, A | 7C | 40 | PS, NM, NP, A | | 4A | 60 |
| V, NM, NP, N | C2 | 42 | PS, NM, NP, N | | 82 | 62 |
| V, NM, PF, A | D7 | 50 | PS, NM, PF, A | | 97 | 70 |
| V, NM, PF, N | D9 | 52 | PS, NM, PF, N | | 99 | 72 |
| V, MA, NP, A | C8 | 48 | PS, MA, NP, A | | 88 | 68 |
| V, MA, NP, N | D1 | 4A | PS, MA, NP, N | | 91 | 6A |
| V, MA, PF, A | E7 | 58 | PS, MA, PF, A | | A7 | 78 |
| V, MA, PF, N | E9 | 5A | PS, MA, PF, N | | A9 | 7A |
| P, NM, NP, A | C1 | 41 | PA, NM, NP, A | 2) | 81 | 61 |
| P, NM, NP, N | C3 | 43 | PA, NM, NP, N | 2) | 83 | 63 |
| P, NM, PF, A | D8 | 51 | PA, NM, PF, A | 2) | 98 | 71 |
| P, NM, PF, N | E2 | 53 | PA, NM, PF, N | 2) | A2 | 73 |
| P, MA, NP, A | C9 | 49 | PA, MA, NP, A | 2) | 89 | 69 |
| P, MA, NP, N | D2 | 4B | PA, MA, NP, N | 2) | 92 | 6B |
| P, MA, PF, A | E8 | 59 | PA, MA, PF, A | 2) | A8 | 79 |
| P, MA, PF, N | BB | 5B | PA, MA, PF, N | 2) | FB | 7B |

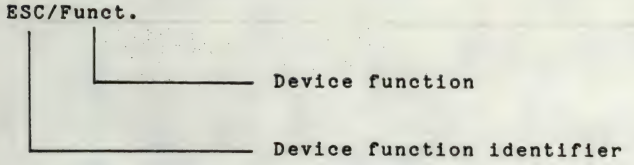
## 3.2 Field attributes at 48 field handling characters per line

| Bit x set to 1 | Field definition | Display attribute |
|---|---|---|
| Bit 1 | Protected fields | Standard font, |
| Bit 6 | Protected fields, sendable | Reduced intensity |
| Bit 1 + 6 | Protected fields; only field address sendable | |
| Bit 2 | Numeric fields | Italics or standard font (adjustable via soldered strap) |
| Bit 3 | Modified field | |
| Bit 5 | Printable field | |
| Bit 4 | No field definition | Flashing |

In the case of 48 field handling characters per line, the display attributes can be set via soldered straps on the pc board as per Section 3.3.2 of the main part of this publication.

## 4 Device function control characters (ESC sequences)

Format of control character sequence (data display terminal controller)




### 4.1 Values for 9750 Data Display Terminal functions

| Function | EBCDIC | ASCII |
|---|---|---|
| • Cursor positioning functions (relative) | | |
| SMR | 7C | 40 |
| SML | C1 | 41 |
| SMO | C2 | 42 |
| SMU | C3 | 43 |
| SNZ | C4 | 44 |
| SZA | C5 | 45 |
| SBA | 82 | 62 |
| SDZ | 92 | 6B |
| TAR | C6 | 46 |
| TAL | C7 | 47 |
| • Text shift and editing functions | | |
| EFG | C8 | 48 |
| AFG | C9 | 49 |
| EFZ | D2 | 4B |
| AFZ | D1 | 4A |
| RU | 4A | 60 |
| • Erase functions | | |
| LZF | D3 | 4C |
| LSP | 84 | 64 |
| LVD | 85 | 65 |
| LVA | 83 | 63 |

Values for 9750 Data Display Terminal (cont.)

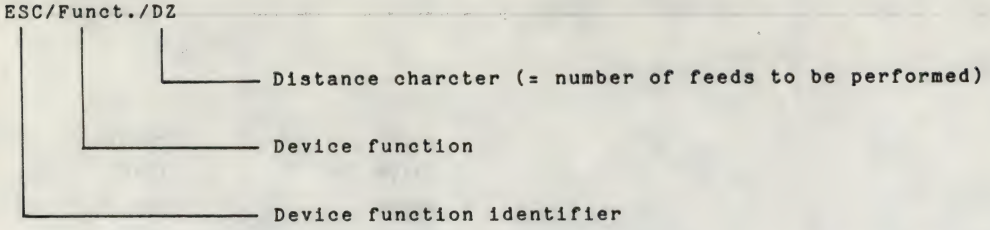
| Function | EBCDIC | ASCII |
|---|---|---|
| • Send functions | | |
| DÜ1 | 86 | 66 |
| DÜ2 | D6 | 51 |
| K1 | E2 | 53 |
| K2 | E3 | 54 |
| K3 | E4 | 55 |
| K4 | E5 | 56 |
| K5 | E6 | 57 |
| K6 | D4 | 4D |
| K7 | D5 | 4E |
| K8 | D6 | 4F |
| K9 | 6F | 3F |
| K10 | 6E | 3E |
| K11 | 7E | 3D |
| K12 | 4C | 3C |
| K13 | 5E | 3B |
| K14 | 7A | 3A |
| F1 | BB | 5B |
| F2 | BC | 5C |
| F3 | BD | 5D |
| F4 | 6A | 5E |
| F5 | 6D | 5F |
| • Intra-system data interchange functions | | |
| LA1 | D8 | 51 |
| LA2 | D9 | 52 |
| LA3 | 40;E7 | 20;58 |
| LA4 | 40;BC | 20;5C |
| LA5 | 40;BD | 20;5D |
| LA6 | 40;6A | 20;5E |
| LA7 | 40;6D | 20;5F |
| • Special functions | | |
| EM | | 19 |
| AKA | F0 | 30 |
| PAR | D7 | 50 |
| MAR | 93 | 6C |
| FAZ | 94 | 6D |
| RS | 95 | 6E |
| VA | 96 | 6F |
| BRS | 97 | 70 |
| LZE | 98 | 71 |

Values for 9750 Data Display Terminal functions (cont.)

| Function | EBCDIC | ASCII |
|---|---|---|
| • Special functions (cont.) | | |
| AM | 40;D7 | 20;50 |
| SS | 40;D9 | 20;52 |
| FON | 40;4A | 20;60 |
| PAK | 40;81 | 20;61 |
| WDH | 40;82 | 20;62 |
| • Font switchover functions (printout) | | |
| Italics | F3 | 33 |
| Standard font | F4 | 34 |
| Wide font | F5 | 36 |
| • Programmable key functions | | |
| P | 91 | 6A |
| P1 | 83 | 63 |
| P2 | 81 | 61 |
| P3 | 88 | 68 |
| P4 | 89 | 69 |
| P5 | 5C | 2A |
| P6 | 4E | 2B |
| P7 | 40;7C | 20;40 |
| P8 | 40;C1 | 20;41 |
| P9 | 40;C2 | 20;42 |
| P10 | 40;C3 | 20;43 |
| P11 | 40;C4 | 20;44 |
| P12 | 40;C5 | 20;45 |
| P13 | 40;C6 | 20;46 |
| P14 | 40;C7 | 20;47 |
| P15 | 40;C8 | 20;48 |
| P16 | 40;C9 | 20;49 |
| P17 | 40;D1 | 20;4A |
| P18 | 40;D2 | 20;4B |
| P19 | 40;D3 | 20;4C |
| P20 | 40;D4 | 20;4D |

## 4.2 Printer terminal control

Format of control character sequence




## 4.3 Values for printer terminal control functions

| Function | EBCDIC | ASCII |
|---|---|---|
| Vertical tab, paper web 1 (VSE1) | F6 | 36 |
| Vertical tab, paper web 2 (VSE2) | F7 | 37 |
| Horizontal tab | F8 | 38 |

## 4.4 Distance character values

| Distance | EBCDIC | ASCII | Distance | EBCDIC | ASCII | Distance | EBCDIC | ASCII | Distance | EBCDIC | ASCII |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 07 | 7F | 25 | 87 | 67 | 49 | D6 | 4F | 73 | F7 | 37 |
| 2 | FF | 7E | 26 | 86 | 66 | 50 | D5 | 4E | 74 | F6 | 36 |
| 3 | FD | 7D | 27 | 85 | 65 | 51 | D4 | 4D | 75 | F5 | 35 |
| 4 | 4F | 7C | 28 | 84 | 64 | 52 | D3 | 4C | 76 | F4 | 34 |
| 5 | FB | 7B | 29 | 83 | 63 | 53 | D2 | 4B | 77 | F3 | 33 |
| 6 | A9 | 7A | 30 | 82 | 62 | 54 | D1 | 4A | 78 | F2 | 32 |
| 7 | A8 | 79 | 31 | 81 | 61 | 55 | C9 | 49 | 79 | F1 | 31 |
| 8 | A7 | 78 | 32 | 4A | 60 | 56 | C8 | 48 | 80 | F0 | 30 |
| 9 | A6 | 77 | 33 | 6D | 5F | 57 | C7 | 47 | 81 | 61 | 2F |
| 10 | A5 | 76 | 34 | 6A | 5E | 58 | C6 | 46 | 82 | 4B | 2E |
| 11 | A4 | 75 | 35 | BD | 5D | 59 | C5 | 45 | 83 | 60 | 2D |
| 12 | A3 | 74 | 36 | BC | 5C | 60 | C4 | 44 | 84 | 6B | 2C |
| 13 | A2 | 73 | 37 | BB | 5B | 61 | C3 | 43 | 85 | 4E | 2B |
| 14 | 99 | 72 | 38 | E9 | 5A | 62 | C2 | 42 | 86 | 5C | 2A |
| 15 | 98 | 71 | 39 | E8 | 59 | 63 | C1 | 41 | 87 | 5D | 29 |
| 16 | 97 | 70 | 40 | E7 | 58 | 64 | 7C | 40 | 88 | 4D | 28 |
| 17 | 96 | 6F | 41 | E6 | 57 | 65 | 6F | 3F | 89 | 7D | 27 |
| 18 | 95 | 6E | 42 | E5 | 56 | 66 | 6E | 3E | 90 | 50 | 26 |
| 19 | 94 | 6D | 43 | E4 | 55 | 67 | 7E | 3D | 91 | 6C | 25 |
| 20 | 93 | 6C | 44 | E3 | 54 | 68 | 4C | 3C | 92 | 5B | 24 |
| 21 | 92 | 6B | 45 | E2 | 53 | 69 | 5E | 3B | 93 | 7B | 23 |
| 22 | 91 | 6A | 46 | D9 | 52 | 70 | 7A | 3A | 94 | 7F | 22 |
| 23 | 89 | 69 | 47 | D8 | 51 | 71 | F9 | 39 | 95 | 5A | 21 |
| 24 | 88 | 68 | 48 | D7 | 50 | 72 | F8 | 38 | 96 | 40 | 20 |

- Note 1

  The following device control characters are assigned their own positions in the code table and are thus not governed by the code extension rules:

  LF, VT, FF, CR, DC1, DC2, DC3, NUL, EM.

- Note 2

  Except for the character for switchover to ZV2, the control characters for font switchover (printout) - see Sections 4.1 and 4.2 of this Appendix - are also used for printer terminal control functions.

  The tables in Sections 4.3 and 4.4 are also used with regard to 9750 control functions (addressing of local printer).

## 5 Use of message headers (host-to-terminal transmission)

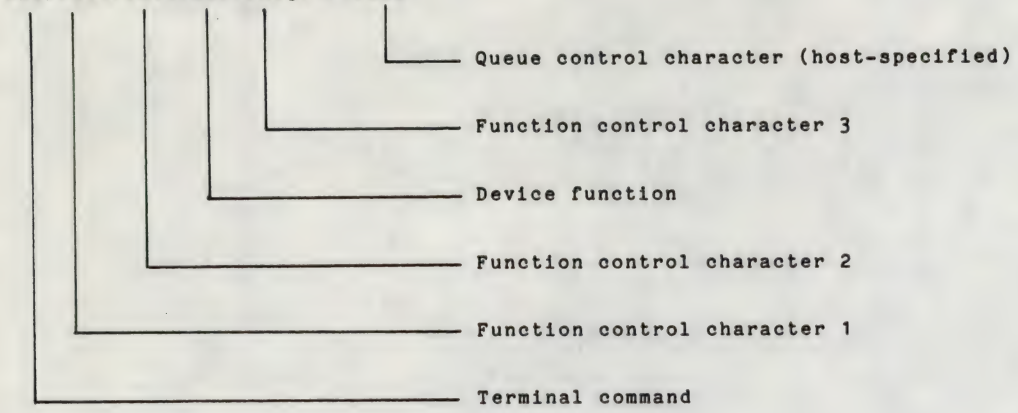
Format of message header

SOM (NBZ)/Parameter range 0/Parameter range 1/Parameter 2

printer (terminal) controller
9750 controller
start-of-message character

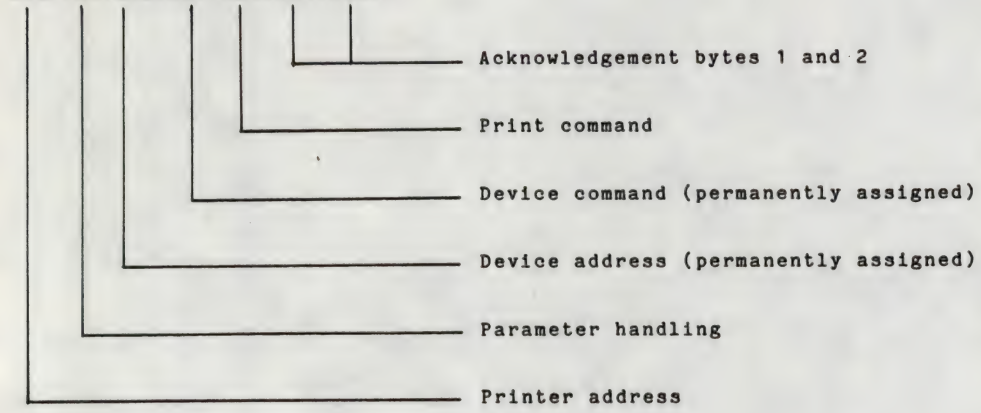
Format of parameter range 0

DAN/FST1/FST2/GEF/FST3/*/*/WAR




Format of parameter range 1

DRA/PBH/GAD/GBF/ADA/RB1/RB2/*




Note on parameter range format

/*/ fields are not occupied at present.

## 5.1 Values for start-of-message character

| Transmission from host to | | Code EBCDIC | Code ASCII |
|---|---|---|---|
| 9750 | No message header follows. | 7C | 40 |
| | 8-byte header (PARAM 0) follows. | C8 | 48 |
| | 16-byte header (PARAM 0 and 1) follows. | D7 | 50 |
| | 24-byte header (PARAM 0, 1 and 2) follows. | E7 | 58 |
| Printer terminal | No message header follows. | 7C | 40 |
| | 8-byte header (PARAM 1) follows. | C8 | 48 |

## 5.2 Values for terminal commands

| Terminal command | EBCDIC | ASCII |
|---|---|---|
| Read unprotected fields. | 7C | 40 |
| Read modified fields. | C1 | 41 |
| Read from cursor. | C2 | 42 |
| Read unprotected fields, without null characters. | C4 | 44 |
| Read modified fields, without null characters. | C5 | 45 |
| Read from cursor, without null characters. | C6 | 46 |

## 5.3 Values for function control character 2

| Function control character 2 (FST2) | EBCDIC | ASCII |
|---|---|---|
| LSP function can be initiated via keyboard; autotab (ATAB) deactivated. | 7C | 40 |
| LSP function can be initiated via keyboard; autotab (ATAB) activated. | C4 | 44 |
| LSP function inhibited via keyboard; autotab (ATAB) deactivated. | C1 | 41 |
| LSP function inhibited via keyboard; autotab (ATAB) activated. | C5 | 45 |

## 5.4 Values for function control character 3

| Function control character 3 (FST3) | EBCDIC | ASCII |
|---|---|---|
| Keyboard released after data input (transmission to host). | 7C | 40 |
| After data input (transmission to host), keyboard remains blocked until data output (transmission from host) is initiated. | C2 | 50 |

## 5.5 Values for printer addresses in a cluster-controller-based configuration

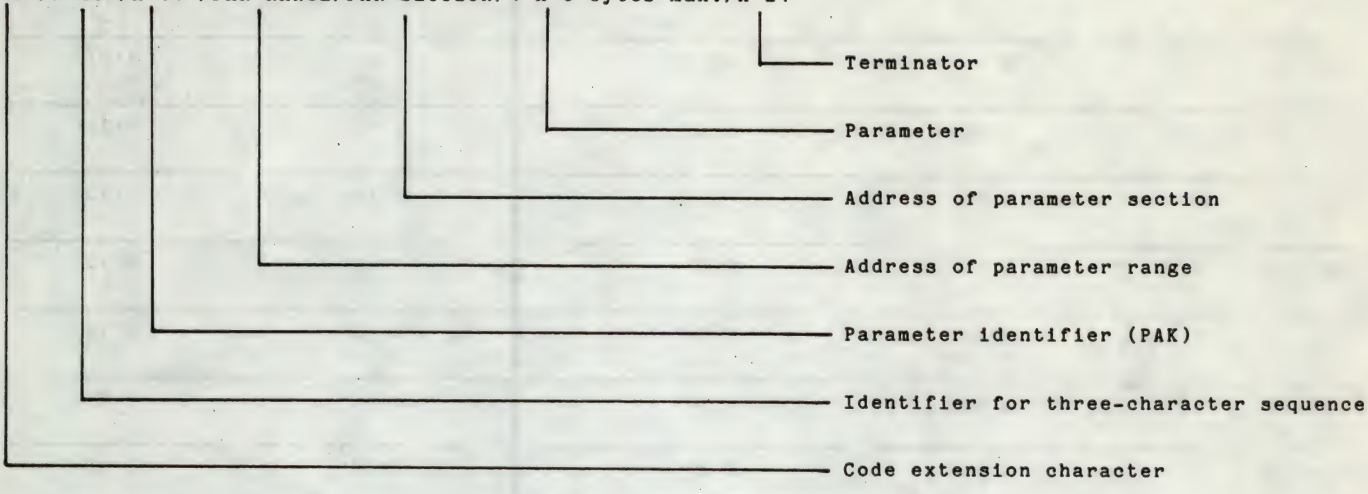
| Printer address | EBCDIC | ASCII | Printer address | EBCDIC | ASCII |
|---|---|---|---|---|---|
| Printer 1 (under printer terminal) on channel 0 of a cluster controller | 40 | 20 | Printer 2 - Channel 16 | D7 | 50 |
| | | | " " 17 | D8 | 51 |
| Printer 1 - Channel 1 | 5A | 21 | " " 18 | D9 | 52 |
| " " 2 | 7F | 22 | " " 19 | E2 | 53 |
| " " 3 | 7B | 23 | " " 20 | E3 | 54 |
| " " 4 | 5B | 24 | " " 21 | E4 | 55 |
| " " 5 | 6C | 25 | " " 22 | E5 | 56 |
| " " 6 | 50 | 26 | " " 23 | E6 | 57 |
| " " 7 | 7D | 27 | " " 24 | E7 | 58 |
| " " 8 | 4D | 28 | " " 25 | E8 | 59 |
| " " 9 | 5D | 29 | " " 26 | E9 | 5A |
| " " 10 | 5C | 2A | " " 27 | BB | 5B |
| " " 11 | 4E | 2B | " " 28 | BC | 5C |
| " " 12 | 6B | 2C | " " 29 | BD | 5D |
| " " 13 | 60 | 2D | " " 30 | 6A | 5E |
| " " 14 | 4B | 2E | " " 31 | 6D | 5F |
| " " 15 | 61 | 2F | Unassigned pr. chann. 0 | 4A | 60 |
| " " 16 | F0 | 30 | " " 1 | 81 | 61 |
| " " 17 | F1 | 31 | " " 2 | 82 | 62 |
| " " 18 | F2 | 32 | " " 3 | 83 | 63 |
| " " 19 | F3 | 33 | " " 4 | 84 | 64 |
| " " 20 | F4 | 34 | " " 5 | 85 | 65 |
| " " 21 | F5 | 35 | " " 6 | 86 | 66 |
| " " 22 | F6 | 36 | " " 7 | 87 | 67 |
| " " 23 | F7 | 37 | " " 8 | 88 | 68 |
| " " 24 | F8 | 38 | " " 9 | 89 | 69 |
| " " 25 | F9 | 39 | " " 10 | 91 | 6A |
| " " 26 | 7A | 3A | " " 11 | 92 | 6B |
| " " 27 | 5E | 3B | " " 12 | 93 | 6C |
| " " 28 | 4C | 3C | " " 13 | 94 | 6D |
| " " 29 | 7E | 3D | " " 14 | 95 | 6E |
| " " 30 | 6E | 3E | " " 15 | 96 | 6F |
| " " 31 | 6F | 3F | " " 16 | 97 | 70 |
| Printer 2 - Channel 0 | 7C | 40 | " " 17 | 98 | 71 |
| " " 1 | C1 | 41 | " " 18 | 99 | 72 |
| " " 2 | C2 | 42 | " " 19 | A2 | 73 |
| " " 3 | C3 | 43 | " " 20 | A3 | 74 |
| " " 4 | C4 | 44 | " " 21 | A4 | 75 |
| " " 5 | C5 | 45 | " " 22 | A5 | 76 |
| " " 6 | C6 | 46 | " " 23 | A6 | 77 |
| " " 7 | C7 | 47 | " " 24 | A7 | 78 |
| " " 8 | C8 | 48 | " " 25 | A8 | 79 |
| " " 9 | C9 | 49 | " " 26 | A9 | 7A |
| " " 10 | D1 | 4A | " " 27 | FB | 7B |
| " " 11 | D2 | 4B | " " 28 | 4F | 7C |
| " " 12 | D3 | 4C | " " 29 | FD | 7D |
| " " 13 | D4 | 4D | " " 30 | FF | 7E |
| " " 14 | D5 | 4E | " " 31 | 07 | 7F |
| " " 15 | D6 | 4F | Local pr. connected to 9750 | 00 | 00 |

## 6 Use of parameter entries

The start-of-message character must contain the information "no message header" ('40').

Format of parameter entries

| Char.1 | 2 | 3 | 4 | 5 | 6, 7, 8, 9, 10 | 11 |
|---|---|---|---|---|---|---|
| X'1B' | X'20' | X'61' | PAR RANGE | PAR SECTION | 4 x 8 bytes max. | X'21' |




### 6.1 Values for parameter range addresses

| | |
|---|---|
| X'40' | Parameter range for text manipulation and function control; entries for system line |
| X'41' - X'47' | Parameter range for terminal user |
| X'48' | Parameter range for host initiation of a device function |
| X'49' - X'4F' | Parameter range for host-controlled LA functions |

### 6.2 Values for parameter section addresses

| | |
|---|---|
| X'40' | Base parameter range PAR 00L/D - PAR 70L/D |
| X'41' | 1st extension: PAR 01L/D, PAR 02L |
| X'42' | 2nd extension: PAR 02D |
| X'43' | 3rd extension: PAR 03D |
| X'50' | Entries for system line |

## 6.3 Values for parameter addresses PAR 00L through PAR 70L

| Parameter | Range Address | | Section Address | |
|---|---|---|---|---|
| | ASCII | EBCDIC | ASCII | EBCDIC |
| PAR 00L | X'40' | X'7C' | X'40' | X'7C' |
| PAR 01L | X'40' | X'7C' | X'41' | X'C1' |
| PAR 10L | X'41' | X'C1' | X'40' | X'7C' |
| PAR 02L | X'41' | X'C1' | X'41' | X'C1' |
| PAR 20L | X'42' | X'C2' | X'40' | X'7C' |
| PAR 30L | X'43' | X'C3' | X'40' | X'70' |
| PAR 40L | X'44' | X'C4' | X'40' | X'7C' |
| PAR 50L | X'45' | X'C5' | X'40' | X'7C' |
| PAR 60L | X'46' | X'C6' | X'40' | X'7C' |
| PAR 70L | X'47' | X'C7' | X'40' | X'7C' |

## 6.4 Values for parameter addresses PAR 00D through PAR 70D

| Parameter | Range Address | | Section Address | |
|---|---|---|---|---|
| | ASCII | EBCDIC | ASCII | EBCDIC |
| PAR 00D | X'48' | X'C8' | X'40' | X'7C' |
| PAR 01D | X'48' | X'C8' | X'41' | X'C1' |
| PAR 02D | X'48' | X'C8' | X'42' | X'C2' |
| PAR 03D | X'48' | X'C8' | X'43' | X'C3' |
| PAR 10D | X'49' | X'C9' | X'40' | X'7C' |
| PAR 20D | X'4A' | X'D1' | X'40' | X'7C' |
| PAR 30D | X'4B' | X'D2' | X'40' | X'7C |
| PAR 40D | X'4C' | X'D3' | X'40' | X'7C' |
| PAR 50D | X'4D' | X'D4' | X'40' | X'7C' |
| PAR 60D | X'4E' | X'D5' | X'40' | X'7C' |
| PAR 70D | X'4F' | X'D6' | X'40' | X'7C' |

## 6.5 Values for parameter addresses for system messages

| Parameter | Range Address | | Section Address | |
|---|---|---|---|---|
| | ASCII | EBCDIC | ASCII | EBCDIC |
| | X'40' | X'7C' | X'50' | X'D7' |

## 6.6 Values for send commands

| Send command | ASCII | EBCDIC |
|---|---|---|
| Send unprotected fields | X'40' | X'7C' |
| Send modified fields | X'41' | X'C1' |
| Send from cursor | X'42' | X'C2' |
| Send display buffer contents | X'43' | X'C3' |
| Send unprotected fields, without null characters | X'44' | X'C4' |
| Send modified fields, without null characters | X'45' | X'C5' |
| Send from cursor, without null characters | X'46' | X'C6' |
| Send from start marker to cursor, without null characters | X'48' | X'C8' |
| Send unprotected fields, with relevant null characters | X'4C' | X'D3' |
| Send modified fields, with relevant null characters | X'4D' | X'D4' |
| Send unprotected fields from cursor, with relevant null characters | X'4E' | X'D5' |
| Send modified fields only, without null characters | X'4F' | X'D6' |
| Send current field only, i.e. the field containing the cursor | X'47' | X'C7' (field mode only) |

## Appendix III Input-message evaluation tables

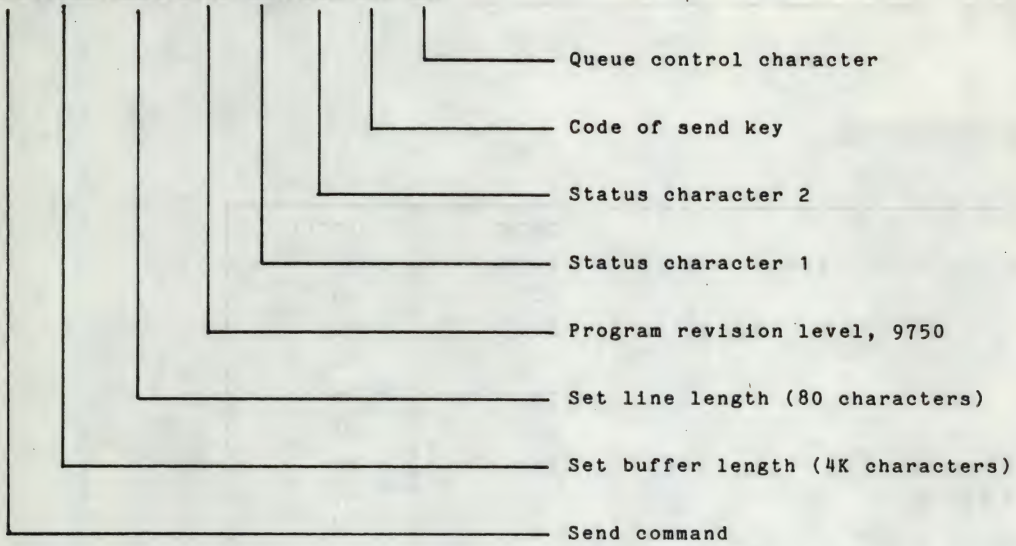
### 1 Message header for data entry (terminal-to-host transmission)

#### 1.1 Format of message header (from terminal)

SOM (NBZ) / PAR 00E / PAR 01E / ...

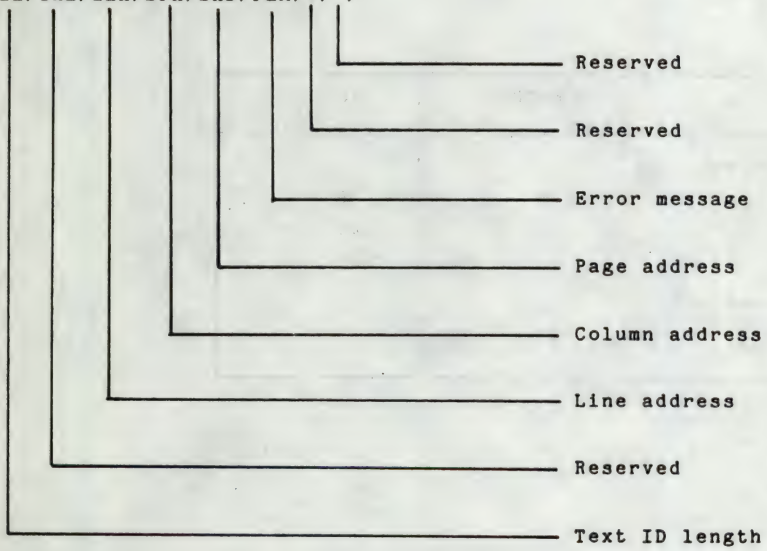
##### 1.1.1 Format of PAR 00E

SAW/PFLNG/ZLNG/AUSG/ZZ1/ZZ2/CDS/WAR




##### 1.1.2 Format of PAR 01E

TIL/TKL/ZLA/SPA/SAD/FEM/*/*/



### 1.1.3 Values for start-of-message character

| ASCII | EBCDIC | Meaning |
|---|---|---|
| X'41' | C1 | Positive acknowledgement |
| X'42' | C2 | Negative acknowledgement |
| X'48' | C8 | PAR 00E included in entry sequence |
| X'50' | D7 | PAR 00E + 01E included in entry seqüence |
| X'58' | E7 | PAR 00E + 01E + 02E included in entry sequence (field mode) |

### 1.2 Values for status character 1

| Keylock switch status | EBCDIC | ASCII |
|---|---|---|
| No key (turned) | 7C | 40 |
| Keylock switch S1 turned | C2 | 42 |
| Keylock switch S2 turned | C4 | 44 |
| Keylock switch S3 turned | C8 | 48 |
| Keylock switches S1 and S2 turned | C6 | 46 |
| Keylock switches S1 and S3 turned | D1 | 4A |
| Keylock switches S2 and S3 turned | D3 | 4C |
| Keylock switches S1, S2 and S3 turned | D5 | 4E |

### 1.3 Values for status character 2

| Connection status | EBCDIC | ASCII |
|---|---|---|
| Neither the keyboard nor the printer is connected or operable. | 7C | 40 |
| Keyboard connected and operable. | C1 | 41 |
| Only the printer is connected and operable. | C2 | 42 |
| Both the keyboard and the printer are connected and operable. | C3 | 43 |

## 1.4 Values for status byte

| Printer status | EBCDIC | ASCII |
|---|---|---|
| No printer errors | D7 | 50 |
| Printer error: | | |
| HALT button pressed (local mode; LB) | C4 | 44 |
| Device error (GF) | C5 | 45 |
| End of paper (PE) | C6 | 46 |
| Format error (FR) | C8 | 48 |
| Printer not connected | 4A | 60 |

Note

For all practical purposes, the print error bits set in the status byte are important only for error diagnosis. Error bit combinations may occur.

## 1.5 Field addresses

- The format of the control character sequence corresponds, with the exception of one item, to that illustrated in Section 1 of Appendix II: ITB is not included.

- The distance value of the page address is always 0.

## Abbreviations

This list comprises the German abbreviations used in both the German and English versions of this User's Guide, each with a corresponding translation and/or explanation, plus a number of relevant English abbreviations accepted as internationally valid designations.

| | |
|---|---|
| AD | Address |
| ADA | Print command |
| ADE | Address character (selecting) |
| ADR | Address character (polling) |
| ASZ | Display control character |
| AUSG | (Program) revision level, 9750 Data Display Terminal |
| BAM | Bit-serial communication protocol for connection to cluster controllers |
| BAST | BAM interface controller |
| BCC | Block check character |
| BK | Mailbox facility |
| CDF | Code of transmission-initiating function |
| CDS | Code of send key |
| CFEP | Compact front-end processor |
| DCE | Data communication equipment |
| DLE | Data link escape character (for switchover to transparent mode) |
| DTF | Data transmission facilities |
| ENQ | Enquiry (polling) |
| ESC | Escape |
| ESP | Switch settings memory |
| ETB | End of transmission block |
| ETX | End of text |
| FBZ | Field handling character |
| FEM | Error message |
| FEP | Front-end processor |
| FKZ | Function identifier |
| FR | Format error |
| FST | Function control character |
| FTZ | Field separator |
| GAD | Device address |
| GBF | Device command |
| GDN | Baseband modem |
| GEF | Device function character |
| GF | Device error |
| IFEP | Integrated front-end processor |
| ITC | Integrated terminal controller |

| | |
|---|---|
| KAN | Channel address |
| KMS | Medium-speed concentrator |
| LAP | LA function parameter |
| LB | Local mode |
| LSV | Low-speed variant (protocol) |
| MA | Markable field |
| MINZL | Minimum cell size |
| MO | Modified fields |
| MSV | Medium-speed variant (protocol) |
| NBZ | Start-of-message character (see SOM) |
| NKA | Message ID byte, host-to-terminal transmission |
| NKE | Message ID byte, terminal-to-host transmission |
| NM | Non-markable fields |
| NP | Non-printable fields |
| PAG | Parameter entries |
| PAK | Parameter identifier character |
| PBH | Parameter handling |
| PDN | Program System for Teleprocessing and Network Control |
| PE | End of paper |
| PF | Printable fields |
| PFLNG | Buffer length |
| RB | Acknowledgement byte |
| RFEP | Remote front-end processor |
| SAD | Page address |
| SAW | Send command |
| SPA | Column address |
| STB | Status byte |
| STX | Start of text |
| TI | Text identification (ID) |
| TIL | Text ID length |
| V | Variable fields |
| WAR | Queue control character |
| ZLA | Line address |
| ZLNG | Line length |
| ZZ | Status character |

**Proposals**

**Corrections**

To
SIEMENS AKTIENGESELLSCHAFT
D ST PM 2
Otto-Hahn-Ring 6

**8000 München 83**

**for Manual:**

**9750 Data Display Terminal TRANSDATA**

**User's Guide**

**Order No. U173-J-Z76-1-7600**
**Edition: March 1983**

From:

Name _____

Company/Department _____

Address _____

_____

Tel.

Printing errors have an uncanny knack of eluding detection. Should you come upon any misprints in reading this manual, will you please notify us on this form. We should also be grateful for any helpful criticisms and suggestions for improvement.

**Proposals and/or corrections:**

Proposals

Corrections

for Manual:

9750 Data Display Terminal
TRANSDATA

User's Guide

Order No. U173-J-Z70-1-7600
Edition: March 1983

Printing errors have an uncanny knack of eluding detection. Should you come upon any misprints in reading this manual, will you please notify us on this form. We should also be grateful for any helpful criticisms and suggestions for improvement.

To
SIEMENS AKTIENGESELLSCHAFT
O ST PM 2
Otto-Hahn-Ring 6
8000 München 83

From:
Name

Company/Department

Address

Tel.

Proposals and/or corrections:

Published by Bereich Datentechnik
Postfach 830951, D-8000 München 83

SIEMENS AKTIENGESELLSCHAFT

Order No. U173-J-Z76-1-7600
Printed in the Federal Republic of Germany
1320 AG 4830.5 (2260)